

PEMBANGUNAN SISTEM INFORMASI MANAJEMEN PERIZINAN DI DINAS KOMINFO KOTA PASURUAN

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Anisah Evi Nur Rohmah

NIM: 135150201111220



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

PENGESAHAN

PEMBANGUNAN SISTEM INFORMASI MANAJEMEN PERIZINAN DI DINAS
KOMINFO KOTA PASURUAN

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Anisah Evi Nur Rohmah
NIM: 135150201111220

Skripsi ini telah diuji dan dinyatakan lulus pada
03 Agustus 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Denny Sagita R., S.Kom, M.Kom
NIP: 19851124 201504 1 001

Nurudin Santoso, S.T., M.T
NIP: 197409162000121001

Mengetahui
Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19710518 200312 1 001

IDENTITAS PENGUJI

Dosen Penguji I

Dosen Penguji II

Fajar Pradana, S.ST, M.Eng
NIP: 19871121 201504 1 004

Tri Astoto Kurniawan, S.T., M.T, Ph.D
NIP: 19710518 2003121 1 001



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 03 Agustus 2018



Anisah Evi Nur Rohmah

NIM: 135150201111220



DAFTAR RIWAYAT HIDUP

Nama : Anisah Evi Nur Rohmah

Tempat Tanggal Lahir : Pasuruan, 15 Juni 1995

Fakultas /Jurusan : Fakultas Ilmu Komputer / Teknik Informatika

Pendidikan :

2001 – 2007 SDN Gentong Pasuruan

2007 – 2010 SMP Negeri 2 Pasuruan

2010 – 2013 SMK Negeri 1 Pasuruan

2013 Universitas Brawijaya

Fakultas Ilmu Komputer / Teknik Informatika

Pengalaman

1. Anggota Cinematic pada tahun 2014
2. Anggota Display atau Pers Mahasiswa 2015-2016
3. Staff Administrasi BPMIF



ABSTRAK

Anisah Evi Nur Rohmah, Pembangunan Sistem Informasi Manajemen Perizinan Di Dinas Kominfo Kota Pasuruan

Pembimbing: Denny Sagita Rusdianto, S.Kom, M.Kom dan Nurudin Santoso, S.T., M.T

Dinas Komunikasi dan Informatika (Kominfo) Kota Pasuruan merupakan badan yang menaungi pembuatan, pemeliharaan dan keamanan data yang berhubungan dengan teknologi informasi. Kominfo melayani pembuatan sistem dalam lingkup pemerintahan Kota Pasuruan. Ada berbagai sistem yang telah dibuat oleh Kominfo salah satunya yaitu sistem website pemerintahan kota, sistem lapor, aplikasi MAJU SDAD dan sistem LPSE. DPMPTSP merupakan salah satu dinas yang ada di Kota Pasuruan yang melayani tentang administrasi perizinan. Pada proses perizinan, DPMPTSP mengalami kendala seperti berkas pemohon yang sering tertukar dengan pemohon lain dan pengajuan izin yang banyak seringkali membutuhkan tingkat ketelitian yang tinggi, sehingga pengurusan izin menjadi lebih lama. Untuk mengatasi hal tersebut, Dinas Kominfo ingin membuat sistem perizinan berbasis website yang bisa diakses secara online. Sistem ini dapat digunakan untuk melakukan pendaftaran perizinan, melihat status perizinan dan memvalidasi berkas perizinan. Ada 4 izin yang akan dibuat dalam sistem ini yaitu izin usaha perdagangan (SIUP), izin tanda daftar perusahaan, izin tanda daftar usaha pariwisata dan izin reklame. Hasil dari analisis kebutuhan yang dilakukan, didapatkan 19 kebutuhan fungsional dan 1 kebutuhan nonfungsional. Pengujian yang dilakukan dalam penelitian ini adalah pengujian *whitebox* dan *blackbox* untuk kebutuhan fungsional. Sedangkan kebutuhan nonfungsional dilakukan pengujian *compatibility*. Berdasarkan hasil pengujian kebutuhan fungsional, didapatkan nilai valid pada setiap kasus uji yang dijalankan yang artinya bahwa sistem tersebut telah sesuai dengan analisis kebutuhan. Untuk hasil dari pengujian *compatibility* juga mendapatkan nilai valid sesuai dengan hasil yang diharapkan.

Kata kunci: kominfo, sistem perizinan, pengujian *whitebox*, pengujian *blackbox*.

ABSTRACT

Anisah Evi Nur Rohmah, Development Of Licensing Management Information System At Kominfo Pasuruan

Adviser: Denny Sagita Rusdianto, S.Kom, M.Kom dan Nurudin Santoso, S.T., M.T

Dinas Komunikasi dan Informatika (KOMINFO) Pasuruan city is a institution that oversees the creation, maintenance and data security that related with information technology. KOMINFO serve making the system within the scope of government in Pasuruan. There are various system that have been made by KOMINFO, one of them is the city government website, reporting system, MAJU SDAD application and LPSE system. DPMPTSP is one of agency in Pasuruan city that serves the administration of licensing. In the licensing process, DPMPTSP encountered several obstacles such as incomplete applicant files that had returned again, the applicant file that was got accidentally exchanged with the other applicants, too many permission application that also require a high degree of accuracy, so making permission is longer. To overcome this, Dinas KOMINFO wants to create a website-based licensing system that can be accessed online. This system will be use to register permissions, view permissions status and validate permissions files. There are 4 permits to be made in this system: SIUP license, TDP license, TDUP license and a advertisement permit. The result of requirement analysis is get 19 functional requirement and 1 nonfunctional requirement. Testing in this study using whitebox testing and blackbox testing for functional requirement. Whereas, the nonfunctional requirements are using compatibility testing. Based on the result of functional requirement testing, the valid value of each test case is performed which means that system is in accordance with requirement analysis. For the result of compatibility testing also get a valid value in accordance with expected results.

Keyword: kominfo, license system, whitebox testing, blackbox testing.

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas limpahan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul “Pembangunan Sistem Informasi Manajemen Perizinan Di Dinas Kominfo Kota Pasuruan”.

Terselesainya penulisan skripsi ini dibantu oleh doa dan motivasi dari berbagai pihak. Untuk itu, penulis menyampaikan terima kasih kepada pihak-pihak berikut.

1. Bapak Denny Sagita R., S.Kom, M.Kom dan Bapak Nurudin Santoso, S.T., M.T selaku dosen pembimbing yang telah membimbing penulis dengan sabar dan memberikan saran yang baik demi terselesainya skripsi ini.
2. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D. selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya.
3. Kedua orang tua tercinta yang memberikan doa dan mencurahkan kasih sayangnya kepada penulis.
4. Bapak Moch Chunaivi S, S.Kom sebagai kepala staff pengembangan aplikasi pada Dinas Kominfo yang bersedia di wawancara dan membantu penulis mendapatkan artifak dokumen perizinan.
5. Hindun Iva NR, adik kembaranku yang baik telah mau membantu dan direpotkan selama pengerjaan skripsi ini.
6. Sahabat penulis Indi, Nuril, Asika, April, Ivarianti yang selalu bersedia direpotkan dan memberikan dukungan kepada penulis.
7. Semua pihak dan teman-teman lainnya yang ikut membantu dalam proses penyelesaian skripsi dan selalu memberikan semangat kepada penulis

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat penulis harapkan. Semoga Allah SWT senantiasa melimpahkan rahmat dan karunia kepada seluruh pihak yang turut membantu penyelesaian skripsi ini. Semoga skripsi ini bermanfaat bagi semua pihak yang menggunakannya.

Malang, 03 Agustus 2018

Penulis

Anisahdenis.1595@gmail.com

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xv
DAFTAR LAMPIRAN	xvii
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Perizinan	5
2.1.1 Pengertian perizinan	5
2.1.2 Fungsi dan Tujuan Perizinan	5
2.1.3 Prosedur Perizinan	6
2.2 Metode Pengembangan Sistem.....	8
2.2.1 <i>Model Waterfall</i>	8
2.3 Pemodelan Berorientasi Objek	11
2.3.1 Diagram <i>Use case</i>	11
2.3.2 Diagram <i>Sequence</i>	12
2.3.3 Diagram Klas.....	13
2.4 Teknologi Pengembangan Sistem.....	14
2.4.1 PHP	14
2.4.2 MySQL	15
BAB 3 METODOLOGI	16

3.1 Studi Pustaka.....	16
3.2 Rekayasa Kebutuhan.....	17
3.3 Perancangan Sistem.....	17
3.4 Implementasi Sistem	18
3.5 Pengujian Sistem.....	18
3.6 Kesimpulan dan Saran	19
BAB 4 REKAYASA KEBUTUHAN.....	20
4.1 Gambaran Umum Sistem.....	20
4.2 Proses Rekayasa Kebutuhan	20
4.2.1 Elisitasi Kebutuhan	20
4.2.2 Identifikasi Aktor	25
4.2.3 Daftar Kebutuhan.....	26
4.3 Pemodelan Kebutuhan	30
4.3.1 Diagram <i>Use case</i>	30
4.3.2 Skenario <i>Use Case</i>	32
BAB 5 PERANCANGAN DAN IMPLEMENTASI.....	47
5.1 Perancangan Sistem.....	47
5.1.1 Perancangan Arsitektur Sistem.....	47
5.1.2 Pemodelan Diagram <i>Sequence</i>	47
5.1.3 Pemodelan Diagram Klas	50
5.1.4 Perancangan Basis Data	54
5.1.5 Perancangan Antarmuka.....	59
5.2 Implementasi Sistem	62
5.2.1 Spesifikasi Sistem	62
5.2.2 Batasan Implementasi.....	62
5.2.3 Implementasi Basis Data	63
5.2.4 Implementasi Kode Program	65
5.2.5 Implementasi Antarmuka	68
BAB 6 PENGUJIAN	73
6.1 Pengujian Unit.....	73
6.1.1 Pengujian Unit Action_siup.....	73
6.1.2 Pengujian Unit Daftar_siup	75

6.1.3 Pengujian Unit Biodata	78
6.1.4 Pengujian Unit Lihatpemohon	79
6.2 Pengujian Integrasi	80
6.2.1 Pengujian Integrasi Update_siup	80
6.2.2 Pengujian Integrasi Validasi_siup.....	83
6.2.3 Pengujian Integrasi Method Form_bio	85
6.2.4 Pengujian Integrasi Method Index()	87
6.3 Pengujian Validasi	89
6.3.1 Pengujian Validasi Registrasi.....	89
6.3.2 Pengujian Validasi <i>Login</i>	90
6.3.3 Pengujian Validasi <i>Login</i> Backoffice	93
6.3.4 Pengujian Validasi <i>Logout</i>	98
6.3.5 Pengujian Validasi Mengupdate Biodata	98
6.3.6 Pengujian Validasi Melihat Notifikasi.....	100
6.3.7 Pengujian Validasi Mengajukan Perizinan SIUP	101
6.3.8 Pengujian Validasi Mengajukan Perizinan TDP	103
6.3.9 Pengujian Validasi Mengajukan Perizinan TDUP	105
6.3.10 Pengujian Validasi Mengajukan Perizinan Reklame	107
6.3.11 Pengujian Validasi Melihat Status Perizinan.....	110
6.3.12 Pengujian Validasi Memeriksa Kelengkapan Berkas.....	110
6.3.13 Pengujian Validasi Membuat SK Perizinan.....	113
6.3.14 Pengujian Validasi Mencetak SK Perizinan	115
6.3.15 Pengujian Validasi Memberikan Nomor Surat.....	117
6.3.16 Pengujian Validasi Menvalidasi Berkas Perizinan	119
6.3.17 Pengujian Validasi Menerbitkan SK Perizinan.....	122
6.3.18 Pengujian Validasi Melihat Daftar Pemohon	124
6.3.19 Pengujian Validasi Menghapus Akun Pemohon	125
6.3.20 Pengujian <i>Compatibility</i>	125
BAB 7 PENUTUP	130
7.1 Kesimpulan.....	130
7.2 Saran	130
DAFTAR PUSTAKA.....	131

LAMPIRAN DOKUMEN PERIZINAN	132
----------------------------------	-----



DAFTAR TABEL

Tabel 2.1 Simbol Diagram <i>Use case</i>	11
Tabel 2.2 Simbol Diagram <i>Sequence</i>	13
Tabel 2.3 Simbol Diagram Klas	13
Tabel 4.1 Kebutuhan Hasil Wawancara	21
Tabel 4.2 Identifikasi Aktor	25
Tabel 4.3 Daftar Kebutuhan Fungsional	27
Tabel 4.4 Daftar Kebutuhan NonFungsional	30
Tabel 4.5 Skenario <i>Use Case</i> Registrasi	32
Tabel 4.6 Skenario <i>Use Case</i> Login	32
Tabel 4.7 Skenario <i>Use Case</i> Login Back Office	33
Tabel 4.8 Skenario <i>Use Case</i> Logout	34
Tabel 4.9 Skenario <i>Use Case</i> Mengupdate Biodata	34
Tabel 4.10 Skenario <i>Use Case</i> Melihat Notifikasi	35
Tabel 4.11 Skenario <i>Use Case</i> Mengajukan Perizinan SIUP	36
Tabel 4.12 Skenario <i>Use Case</i> Mengajukan Perizinan TDP	37
Tabel 4.13 Skenario <i>Use Case</i> Mengajukan Perizinan TDUP	38
Tabel 4.14 Skenario <i>Use Case</i> Mengajukan Perizinan Reklame	39
Tabel 4.15 Skenario <i>Use Case</i> Melihat Status Perizinan SIUP, TDP, TDUP, Reklame	40
Tabel 4.16 Skenario <i>Use Case</i> Memeriksa Kelengkapan Berkas SIUP, TDP, TDUP, Reklame	41
Tabel 4.17 Skenario <i>Use Case</i> Membuat SK Perizinan SIUP, TDP, TDUP, Reklame	41
Tabel 4.18 Skenario <i>Use Case</i> Mencetak SK Perizinan SIUP, TDP, TDUP, Reklame	42
Tabel 4.19 Skenario <i>Use Case</i> Memberikan Nomor Surat	43
Tabel 4.20 Skenario <i>Use Case</i> Menvalidasi Berkas Perizinan	43
Tabel 4.21 Skenario <i>Use Case</i> Menerbitkan SK perizinan SIUP, TDP, TDUP, Reklame	44
Tabel 4.22 Skenario <i>Use Case</i> Melihat Daftar Pemohon	45
Tabel 4.23 Skenario <i>Use Case</i> Menghapus Akun Pemohon	45

Tabel 5.1 Algoritme Action_siup	52
Tabel 5.2 Algoritme Daftar_siup	53
Tabel 5.3 Algoritme Biodata.....	54
Tabel 5.4 Algoritme Lihatpemohon	54
Tabel 5.5 Spesifikasi Perangkat Keras Komputer	62
Tabel 5.6 Spesifikasi Perangkat Lunak	62
Tabel 5.7 Kode Program Action_siup.....	65
Tabel 5.8 Kode Program Daftar_siup	67
Tabel 5.9 Kode Program Biodata	67
Tabel 5.10 Kode Program Lihatpemohon	68
Tabel 6.1 Hasil Pengujian Unit dari Method Action_siup	75
Tabel 6.2 Hasil Pengujian Unit dari Method Daftar_siup	77
Tabel 6.3 Hasil Pengujian Unit Method Biodata	78
Tabel 6.4 Hasil Pengujian Unit Method Lihatpemohon.....	80
Tabel 6.5 Hasil Pengujian Integrasi Method Update_siup.....	82
Tabel 6.6 Hasil Pengujian Integrasi Method Validasi_siup	84
Tabel 6.7 Hasil Pengujian Integrasi Method Form_bio.....	86
Tabel 6.8 Hasil Pengujian Integrasi Method Lihatpemohon.....	89
Tabel 6.9 Kasus Uji Registrasi dengan Data Lengkap	89
Tabel 6.10 Kasus Uji Registrasi dengan Data Kosong.....	90
Tabel 6.11 Kasus Uji Berhasil <i>Login</i> Sebagai Pengguna	90
Tabel 6.12 Kasus Uji <i>Username</i> dan <i>Password</i> Kosong.....	91
Tabel 6.13 Kasus Uji <i>Username</i> dan <i>Password</i> Tidak Sesuai.....	91
Tabel 6.14 Kasus Uji <i>Username</i> Salah	92
Tabel 6.15 Kasus Uji <i>Password</i> Salah	92
Tabel 6.16 Kasus Uji <i>Login</i> Sebagai Staff <i>Back office</i>	93
Tabel 6.17 Kasus Uji <i>Login</i> Sebagai Kepala Pengendalian.....	93
Tabel 6.18 Kasus Uji <i>Login</i> Sebagai Kepala Bidang	94
Tabel 6.19 Kasus Uji <i>Login</i> Sebagai Kepala Admin	94
Tabel 6.20 Kasus Uji <i>Login</i> Sebagai Kepala Dinas.....	95
Tabel 6.21 Kasus Uji <i>Username</i> dan <i>Password</i> Kosong.....	95
Tabel 6.22 Kasus Uji <i>Username</i> dan <i>Password</i> Tidak Sesuai.....	96

Tabel 6.23 Kasus Uji <i>Username</i> Salah	96
Tabel 6.24 Kasus Uji <i>Password</i> Salah	97
Tabel 6.25 Kasus Uji Level Salah.....	97
Tabel 6.26 Kasus Uji <i>Logout</i>	98
Tabel 6.27 Kasus Uji Update Biodata Tanpa Data Kosong	98
Tabel 6.28 Kasus Uji Edit Biodata Ada Data Kosong	99
Tabel 6.29 Kasus Uji Update Biodata Tanpa Upload Foto	99
Tabel 6.30 Kasus Uji Ada Notifikasi	100
Tabel 6.31 Kasus Uji Tidak Ada Notifikasi	100
Tabel 6.32 Kasus Uji Data SIUP Lengkap	101
Tabel 6.33 Kasus Uji Pengajuan Izin SIUP dengan Data tidak Lengkap.....	102
Tabel 6.34 Kasus Uji Pengajuan Izin SIUP Tanpa Upload Berkas	102
Tabel 6.35 Kasus Uji Pengajuan Izin TDP dengan Data Lengkap.....	103
Tabel 6.36 Kasus Uji Pengajuan Izin TDP dengan Data tidak Lengkap.....	104
Tabel 6.37 Kasus Uji Pengajuan Izin TDP tanpa Upload Berkas	104
Tabel 6.38 Kasus Uji Pengajuan Izin TDUP dengan Data Lengkap	105
Tabel 6.39 Kasus Uji Pengajuan Izin TDUP dengan Data Tidak Lengkap	106
Tabel 6.40 Kasus Uji Pengajuan Izin TDUP tanpa Upload Berkas	106
Tabel 6.41 Kasus Uji Pengajuan Izin Reklame dengan Data Lengkap	107
Tabel 6.42 Kasus Uji Pengajuan Izin Reklame dengan Data tidak Lengkap	108
Tabel 6.43 Kasus Uji Pengajuan Izin Reklame Tanpa Upload Berkas.....	109
Tabel 6.44 Kasus Uji Melihat Status Perizinan	110
Tabel 6.45 Kasus Uji Memeriksa Kelengkapan Berkas SIUP	110
Tabel 6.46 Kasus Uji Memeriksa Kelengkapan Berkas TDP	111
Tabel 6.47 Kasus Uji Memeriksa Kelengkapan Berkas TDUP	112
Tabel 6.48 Kasus Uji Memeriksa Kelengkapan Berkas Reklame	112
Tabel 6.49 Kasus Uji membuat SK perizinan SIUP.....	113
Tabel 6.50 Kasus Uji membuat SK perizinan TDP.....	114
Tabel 6.51 Kasus Uji membuat SK perizinan TDUP	114
Tabel 6.52 Kasus Uji membuat SK perizinan Reklame	115
Tabel 6.53 Kasus Uji Mencetak SK Perizinan SIUP	115
Tabel 6.54 Kasus Uji Mencetak SK Perizinan TDP	116

Tabel 6.55 Kasus Uji Mencetak SK Perizinan TDUP.....	116
Tabel 6.56 Kasus Uji Mencetak SK Perizinan Reklame.....	117
Tabel 6.57 Kasus Uji Memberikan Nomor Surat SIUP	117
Tabel 6.58 Kasus Uji Memberikan Nomor Surat TDP	118
Tabel 6.59 Kasus Uji Memberikan Nomor Surat TDUP	118
Tabel 6.60 Kasus Uji Memberikan Nomor Surat Reklame	119
Tabel 6.61 Kasus Uji Menvalidasi Berkas Perizinan SIUP	119
Tabel 6.62 Kasus Uji Menvalidasi Berkas Perizinan TDP	120
Tabel 6.63 Kasus Uji Menvalidasi Berkas Perizinan TDUP	121
Tabel 6.64 Kasus Uji Menvalidasi Berkas Perizinan Reklame	121
Tabel 6.65 Kasus Uji Menerbitkan SK Perizinan SIUP	122
Tabel 6.66 Kasus Uji Menerbitkan SK Perizinan TDP	123
Tabel 6.67 Kasus Uji Menerbitkan SK Perizinan TDUP.....	123
Tabel 6.68 Kasus Uji Menerbitkan SK Perizinan Reklame.....	124
Tabel 6.69 Kasus Uji Melihat Daftar Pemohon	124
Tabel 6.70 Kasus Uji Menghapus Akun Pemohon.....	125
Tabel 6.71 Hasil Pengujian <i>Compatibility</i>	126



DAFTAR GAMBAR

Gambar 2.1 Flowchart Alur Perizinan	7
Gambar 2.2 Model <i>Waterfall</i>	8
Gambar 3.1 Kerangka Penelitian.....	16
Gambar 4.1 <i>Work Breakdown Structure</i>	22
Gambar 4.2 Proses Bisnis Perizinan Yang digunakan.....	23
Gambar 4.3 Proses Bisnis Perizinan Sistem Yang Baru	24
Gambar 4.4 Aturan Penomoran Spesifikasi Kebutuhan	26
Gambar 4.5 Diagram <i>Use Case</i> Sistem Perizinan	31
Gambar 5.1 Perancangan Arsitektur Sistem Perizinan	47
Gambar 5.2 Pengajuan Izin SIUP.....	48
Gambar 5.3 Menvalidasi Berkas SIUP	49
Gambar 5.4 Update Biodata	49
Gambar 5.5 Menghapus Pemohon	50
Gambar 5.6 Diagram Klas Sistem Perizinan	51
Gambar 5.7 <i>Entity Relationship Diagram</i>	56
Gambar 5.8 <i>Conceptual Data Model</i>	57
Gambar 5.9 <i>Physical Data Model</i>	58
Gambar 5.10 Perancangan Antarmuka Halaman Form SIUP.....	59
Gambar 5.11 Perancangan Antarmuka Halaman Validasi	60
Gambar 5.12 Perancangan Antarmuka Halaman Update Biodata	61
Gambar 5.13 Perancangan Antarmuka Halaman Menghapus Pemohon.....	61
Gambar 5.14 Implementasi Basis Data	64
Gambar 5.15 Halaman Form Pengajuan SIUP (1)	68
Gambar 5.16 Halaman Form Pengajuan SIUP (2)	69
Gambar 5.17 Halaman Form Pengajuan SIUP (3)	69
Gambar 5.18 Halaman Form Pengajuan SIUP (4)	69
Gambar 5.19 Halaman Daftar Pemohon SIUP	70
Gambar 5.20 Halaman Detail Pemohon (1)	70
Gambar 5.21 Halaman Detail Pemohon (2)	70
Gambar 5.22 Halaman Form Cetak SIUP	71
Gambar 5.23 Halaman Update Biodata	71

Gambar 5.24 Halaman Hapus Pemohon.....	72
Gambar 6.1 Pembentukan Node Algoritme Action_siup	74
Gambar 6.2 <i>Flowgraph</i> Action_siup	74
Gambar 6.3 Pembentukan Node Algoritme Daftar_siup.....	76
Gambar 6.4 <i>Flowgraph</i> Daftar_siup.....	76
Gambar 6.5 Pembentukan <i>Node</i> Algoritme Biodata	78
Gambar 6.6 <i>Flowgraph</i> Biodata	78
Gambar 6.7 Pembentukan Node Method Lihatpemohon	79
Gambar 6.8 <i>Flowgraph</i> Lihatpemohon	79
Gambar 6.9 Pembentukan <i>Node</i> Algoritme Update_siup	81
Gambar 6.10 <i>Flowgraph</i> Method Update_siup	81
Gambar 6.11 Pembentukan <i>Node</i> Validasi_siup	83
Gambar 6.12 <i>Flowgraph</i> Validasi_siup.....	84
Gambar 6.13 Pembentukan <i>Node</i> Method Form_bio	85
Gambar 6.14 <i>Flowgraph</i> Method Form_bio	86
Gambar 6.15 Pembentukan <i>Node</i> Method Index()	88
Gambar 6.16 <i>Flowgraph</i> Method Index()	88
Gambar 6.17 Sistem Perizinan Pada <i>Browser Chrome</i>	128
Gambar 6.18 Sistem Perizinan Pada <i>Browser Mozilla Firefox</i>	128
Gambar 6.19 Sistem Perizinan Pada <i>Browser Internet Explorer</i>	129

BAB 1 PENDAHULUAN

1.1 Latar belakang

Izin sering disebut sebagai pernyataan mengabulkan. Menurut Prof. Bagirmanan (2012) mengatakan bahwa izin adalah suatu persetujuan yang dilakukan penguasa yang didasarkan pada peraturan perundang-undangan untuk menyetujui suatu tindakan atau perbuatan yang sebenarnya dilarang. Izin dapat berupa dispensasi, lisensi dan konsesi. Izin digunakan untuk melakukan tindakan atau perbuatan yang dilarang. Fungsi izin yaitu untuk memberikan kepastian hukum bagi pemohon dan masyarakat, sebagai tindakan preventif untuk menghadapi pihak-pihak yang mengganggu dan sebagai pengaman secara hukum (Akbar, et al., 2012).

Ada berbagai jenis izin yang ada dalam pemerintahan seperti izin mendirikan bangunan (IMB), izin usaha perdagangan, izin reklame, izin perluasan, izin usaha perjalanan wisata, izin gangguan, izin usaha rekreasi dan hiburan umum, izin obyek wisata, izin usaha jasa konstruksi, izin hotel dan penginapan dll. (Pusat Kajian, 2011). Dalam instansi pemerintah Kota Pasuruan, Dinas Penanaman Modal dan Pelayanan Terpadu Satu Pintu (DPMPTSP) merupakan sebuah instansi yang melayani administrasi perizinan sesuai dengan peraturan walikota Pasuruan nomor 68 tahun 2011 pasal 2 tentang tugas DPMPTSP yaitu merumuskan dan melaksanakan kebijakan daerah di bidang penanaman modal dan melaksanakan koordinasi dan menyelenggarakan pelayanan administrasi di bidang perizinan secara terpadu.

Dalam melakukan pengurusan izin pada DPMPTSP memerlukan banyak orang dengan melibatkan beberapa bagian dalam setiap proses pembuatan izin. Proses tersebut dimulai dari pengajuan permohonan, pemeriksaan berkas pemohon, validasi berkas pemohon, pemeriksaan lokasi pemohon, pembayaran pajak bagi pemasangan reklame hingga pembuatan dan penerbitan Surat Keputusan (SK) perizinan. Sehingga diperlukan tingkat ketelitian untuk pemrosesan penerbitan SK perizinan tersebut.

Banyaknya pemohon yang mengajukan perizinan, mengakibatkan kesulitan dalam hal pemeriksaan data pemohon. Data pemohon yang bercampur dengan berkas pemohon lain mengakibatkan adanya perpanjangan waktu dalam pengurusan izin. Untuk meningkatkan layanan dan mempermudah pengurusan izin sesuai dengan visi DPMPTSP yaitu "Terwujudnya pelayanan prima di bidang penanaman modal dan pelayanan terpadu satu pintu yang berdaya saing" dan misi yaitu meningkatkan sumberdaya, sarana dan prasarana bidang penanaman modal dan perizinan, memberikan pelayanan yang mudah, murah, cepat, tepat dan *akuntable* di bidang penanaman modal dan perizinan, meningkatkan efektifitas promosi dan informasi penanaman modal dan perizinan, meningkatkan kesadaran masyarakat / pengusaha untuk mematuhi ketentuan-ketentuan bidang penanaman modal dan perizinan, meneliti dan mengembangkan investasi dan pelayanan perizinan serta mendorong tumbuhnya investasi (dpmptsp, 2018). Dari uraian visi dan misi dalam meningkatkan pelayanan yang mudah, murah, cepat dan tepat maka Dinas Komunikasi, Informatika dan Statistika (Kominfo) Kota Pasuruan yang menaungi pembuatan, pemeliharaan dan keamanan data yang berhubungan dengan teknologi informasi ingin membuat sistem perizinan untuk mewujudkan visi dan misi dari Dinas DPMPTSP.

Selain itu, Dinas Kominfo juga ingin meningkatkan layanan *e-government* menjadi lebih baik lagi sesuai dengan kebijakan pemerintah Indonesia tahun 2003 yang telah mengeluarkan

Inpres no 3 tahun 2003 tentang kebijakan dan strategi nasional pengembangan *e-government* yang menginstruksikan kepada setiap kepala daerah untuk mengambil langkah konkret sesuai dengan tugas, fungsi dan kewenangannya untuk melakukan pengembangan *e-government*. Pengembangan *e-government* sendiri merupakan upaya pemerintahan yang berbasis elektronik dalam rangka meningkatkan kualitas layanan public secara efektif dan efisien (Inpres, 2003).

Berdasarkan permasalahan tersebut judul yang diambil adalah “Pembangunan Sistem Informasi Manajemen Perizinan Di Dinas Kominfo Kota Pasuruan”. Penelitian ini diharapkan dapat meningkatkan layanan dan mewujudkan visi dan misi dari Dinas DPMPTSP Kota Pasuruan.

1.2 Rumusan masalah

Berdasarkan uraian latar belakang tersebut, rumusan masalah dalam penelitian ini yaitu:

1. Apa kebutuhan fungsional dan nonfungsional dari SIUP, TDP, TDUP dan reklame yang tepat untuk diimplementasikan ke sistem?
2. Bagaimana hasil perancangan dan implementasi dari sistem perizinan (SIUP, TDP, TDUP dan Reklame)?
3. Bagaimana hasil dari pengujian sistem perizinan (SIUP, TDP, TDUP dan Reklame)?

1.3 Tujuan

Tujuan penelitian sistem perizinan yaitu untuk memudahkan pemohon dalam mengurus perizinan dan meminimalisir terjadinya kehilangan berkas yang menyebabkan waktu pengurusan perizinan menjadi lebih lama.

1.4 Manfaat

Hasil dari penelitian ini diharapkan dapat memberikan manfaat untuk berbagai pengguna. Adapun manfaat dari penelitian ini yaitu:

1. Mempermudah masyarakat dalam mengurus perizinan.
2. Mempermudah masyarakat untuk mengetahui proses perizinan yang dilakukan sudah sampai tahap apa.
3. Menghemat waktu untuk mengurus perizinan.

1.5 Batasan masalah

Batasan-batasan yang digunakan dalam penelitian ini yaitu:

1. Perizinan yang ada pada penelitian ini adalah SIUP, TDP, Reklame, TDUP yang terdiri dari TDUP makanan dan TDUP wisata.
2. Studi kasus penelitian dilakukan pada Dinas Komunikasi dan Informatika di Kota Pasuruan.
3. Penelitian ini berfokus pada pendaftaran izin SIUP, TDP, TDUP dan Reklame.
4. Penelitian ini hanya sampai pada tahap cetak dokumen Surat Keputusan.
5. Sistem perizinan yang dibangun pada penelitian ini merupakan sistem berbasis web.

6. Format form pengajuan izin disesuaikan seperti format yang digunakan di DPMPSTSP Kota Pasuruan.

1.6 Sistematika pembahasan

Sistematika penulisan penelitian ditunjukkan untuk memberikan gambaran dan uraian yang dibagi dalam beberapa Bab meliputi:

BAB I PENDAHULUAN

Menguraikan latar belakang masalah, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan.

BAB II LANDASAN KEPUSTAKAAN

Menguraikan dasar teori yang mendasari tentang teori-teori yang digunakan pada penelitian sistem perizinan.

BAB III METODOLOGI

Menguraikan tentang metode dan langkah kerja yang digunakan dalam penelitian ini, seperti studi literatur, metode analisis kebutuhan, perancangan sistem, implementasi, pengujian sistem, kesimpulan dan saran.

BAB IV ANALISIS KEBUTUHAN

Menguraikan analisis kebutuhan sistem yang di representasikan dengan gambaran umum sistem, identifikasi aktor, analisis kebutuhan fungsional dan non fungsional, diagram *use case* dan skenario *use case*.

BAB V PERANCANGAN DAN IMPLEMENTASI

Menguraikan perancangan perangkat lunak berdasarkan analisis kebutuhan perangkat lunak yang telah di rancang di Bab sebelumnya. Perancangan yang dibuat meliputi pemodelan diagram *sequence* dan diagram klas. Bab ini juga membahas implementasi sistem perizinan.

BAB VI PENGUJIAN

Menguraikan mengenai pengujian yang dilakukan dalam penelitian ini yaitu pengujian *blackbox* dan *whitebox*.

BAB VI PENUTUP

Menguraikan kesimpulan dan saran dari penelitian “Pembangunan Sistem Perizinan Pada Dinas Kementerian Komunikasi dan Informatika (Studi Kasus : Dinas Kominfo Pasuruan)”.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Perizinan

2.1.1 Pengertian perizinan

Izin adalah dispensasi yang dikeluarkan oleh pejabat pemerintahan yang bersangkutan untuk memperbolehkan suatu hal yang sebenarnya dilarang untuk dapat dilaksanakan sesuai dengan persyaratan dan ketentuan yang berlaku (Akbar, et al., 2012). Pemerintah menggunakan izin sebagai sarana yuridis untuk melindungi dan mengamati perilaku warga negaranya agar tidak melakukan hal yang dilarang serta untuk menjaga keseimbangan kepentingan antara masyarakat dengan lingkungannya. Dengan diberlakukannya izin maka pejabat yang bersangkutan memperbolehkan pemohon melakukan tindakan-tindakan tertentu yang dilarang dengan adanya pengawasan demi kepentingan umum.

Dari pemahaman izin yang telah diuraikan, pendapat perizinan menurut Sutedi (2011) bahwa perizinan merupakan suatu bentuk pelaksanaan fungsi yang dimiliki pemerintah untuk memantau kegiatan-kegiatan yang dilakukan masyarakat yang bersifat pengendalian. Ada berbagai bentuk perizinan seperti pendaftaran, rekomendasi, penentuan kuota, sertifikasi dan izin melakukan usaha oleh perusahaan dalam suatu organisasi.

Secara yuridis berdasarkan Peraturan Menteri Dalam Negeri nomor 20 tahun 2008 pasal 1 angka 8 dan 9 tentang Pedoman Organisasi Dan Tata Kerja Unit Pelayanan Perizinan Terpadu Daerah pengertian izin adalah izin adalah dokumen yang dikeluarkan oleh pemerintah daerah berdasarkan peraturan daerah atau peraturan lain yang merupakan bukti legalitas, dinyatakan sah atau diperbolehkannya seseorang atau badan untuk melakukan usaha atau kegiatan tertentu. Sedangkan perizinan adalah pemberian legalitas kepada seseorang atau pelaku usaha/ kegiatan tertentu, baik dalam bentuk izin maupun tanda daftar usaha.

2.1.2 Fungsi dan Tujuan Perizinan

Izin dapat disebut sebagai instrument yuridis dikarenakan pemerintah dapat mengontrol masyarakat atau warga negaranya melalui izin. Ada tiga fungsi izin menurut (Akbar, et al., 2012) yaitu:

1. Dengan adanya izin pemohon dan masyarakat memperoleh kepastian hukum.
2. Izin dapat digunakan sebagai tindakan preventik untuk melawan oknum-oknum yang mengganggu.
3. Izin dapat digunakan sebagai pengaman secara hukum.

Dari fungsi perizinan tersebut, perizinan juga memiliki tujuan yaitu agar pemerintah dapat mempengaruhi masyarakat untuk mengikuti keinginannya. Selain itu menurut Sutedi (2011) tujuan izin digunakan untuk mengarahkan pada aktivitas-aktivitas tertentu seperti izin gangguan atau HO, izin mendirikan bangunan, izin perdagangan dll, mencegah bahaya pada lingkungan, digunakan untuk melindungi objek tertentu, digunakan untuk membagi wilayah, lahan atau benda-benda untuk kebutuhan masyarakat dan memberikan arahan dengan melakukan penyeleksian kepada aktivitas-aktivitas dan orang-orang tertentu.

2.1.3 Prosedur Perizinan

Sesuai dengan peraturan daerah Kota Pasuruan nomor 26 tahun 2011 bahwa ditetapkan pelayanan perizinan pada Dinas Penanaman Modal dan Pelayanan Perijinan Terpadu Satu Pintu (DPMPTSP). Prosedur Perizinan yang terdapat pada Dinas DPMPTSP dijelaskan pada Gambar 2.1

1. *Front Office*

Pada tahap ini pemohon melakukan pengajuan izin dengan meminta formulir perizinan sesuai dengan izin yang diajukan. Pemohon mengisi formulir dan melengkapi berkas. Bagian *front office* menerima formulir dan berkas pemohon untuk diperiksa kelengkapan berkas apakah berkas tersebut dapat diterima atau tidak.

2. *Back Office*

Pada tahap ini bagian back office melakukan verifikasi kelengkapan berkas pemohon.

3. Subbidang Koordinasi dan penelitian lapangan

Pada subbidang koordinasi dan penelitian lapangan akan melakukan pemeriksaan lapangan dan merekomendasikan teknis.

4. *Back Office*

Pada tahap ini back office telah menerima rekomendasi dari koordinasi dan penelitian lapangan apabila izin tersebut harus melakukan survei lapangan. Kemudian back office membuat draft SK perizinan.

5. Kepala sub bidang dan kepala bidang

Pada tahap ini melakukan proses verifikasi draft dari SK yang telah dibuat oleh bagian *back office*.

6. Kepala Dinas

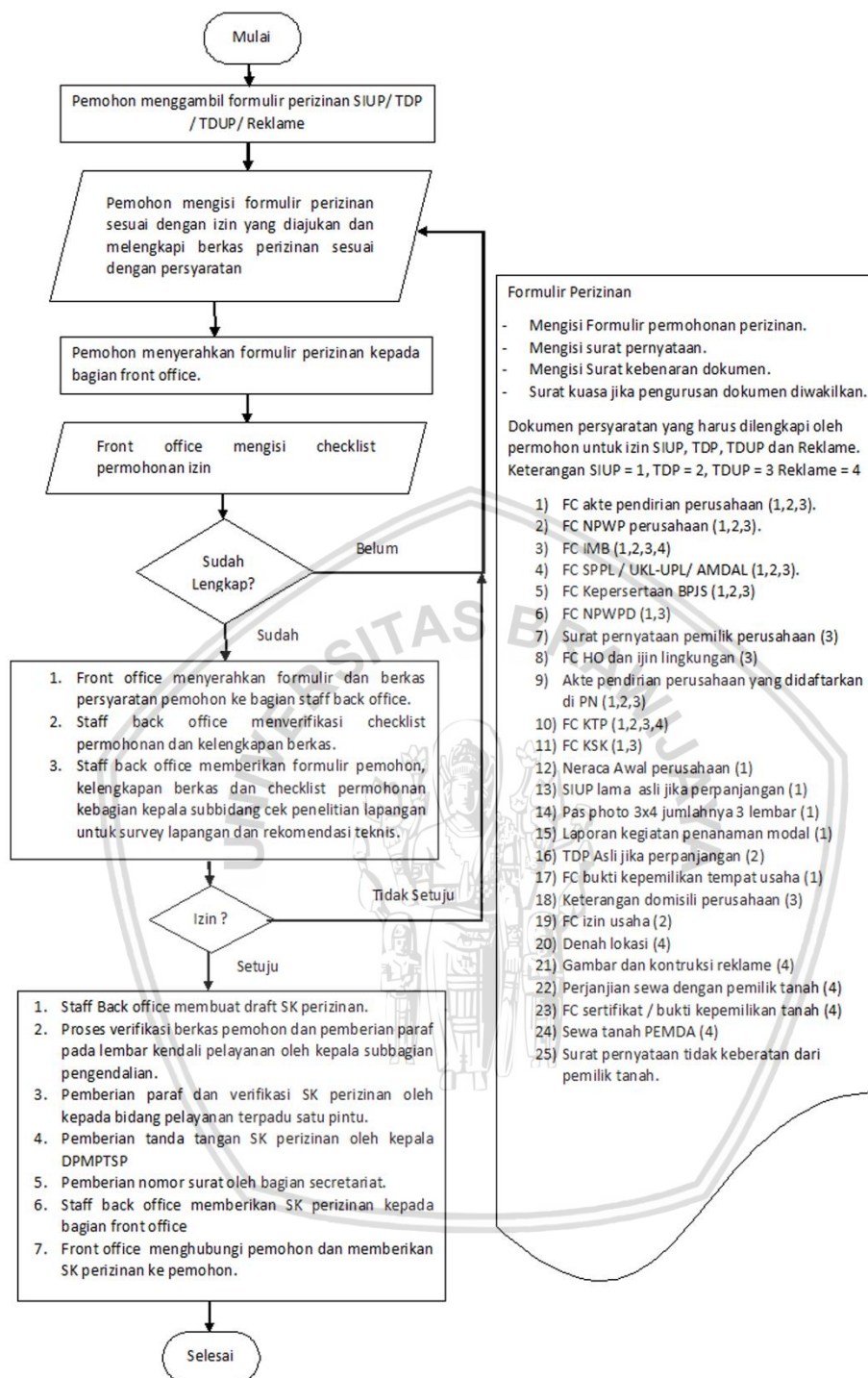
Pada tahap ini Kepala dinas bertugas untuk memeriksa kembali dokumen SK dan melakukan ACC atau menyetujui SK permohonan izin. Jika SK telah di ACC maka SK dapat diberikan kepada pemohon.

7. Sekretariat

Pada tahap ini SK telah selesai dibuat dan telah disetujui oleh kepala dinas DPMPTSP yang selanjutnya bagian sekretariat memberikan nomor surat pada SK tersebut.

8. Front Office

Pada tahap ini front office menghubungi dan memberikan SK kepada pemohon.



Gambar 2.1 Flowchart Alur Perizinan

Sumber: Chunaivy (2017)

2.2 Metode Pengembangan Sistem

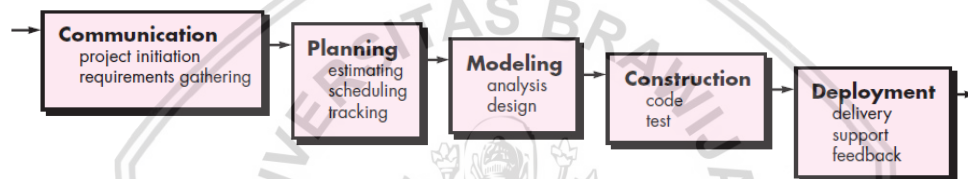
Metode pengembangan sistem merupakan suatu metode yang digunakan dalam melakukan pengembangan sistem perangkat lunak. Tahapan dalam pengembangan sistem disebut *System Development Life Cycle* atau SDLC. SDLC merupakan tahapan atau proses yang digunakan oleh pengembang perangkat lunak dalam membangun sebuah sistem. SDLC

merupakan keseluruhan proses dalam membangun dan mengembangkan sistem melalui beberapa tahapan. Tujuan SDLC yaitu untuk membantu memonitor, mengontrol, mendokumentasikan dari setiap progress yang sedang berjalan dalam proses pembangunan sistem.

Adapun tahapan SDLC menurut Sommerville (2011) yang pertama adalah spesifikasi perangkat lunak dilanjutkan dengan perancangan dan implementasi perangkat lunak, validasi perangkat lunak dan evolusi perangkat lunak. Sedangkan untuk model perangkat lunak yang sering digunakan adalah model *waterfall*, model V, model inkremental, model proses evolusioner, model spiral dan iteratif (Pressman, 2010). Dalam pembangunan sistem perizinan, model SDLC yang digunakan yaitu model air terjun atau *waterfall*.

2.2.1 Model Waterfall

Waterfall merupakan salah satu model pengembangan perangkat lunak. Model *waterfall* ini berkembang secara sistematis dari satu tahap ke tahap lain. Tahapan-tahapan yang ada pada model *waterfall* ini meliputi *communication*, *planning*, *modeling*, *contruction*, deployment (Pressman, 2010). Model *waterfall* ditunjukkan pada Gambar 2.1



Gambar 2.2 Model Waterfall

Sumber: Pressman (2010)

Berikut ini merupakan penjelasan dari setiap tahap yang digambarkan pada gambar 2.2:

1. Komunikasi (*Communication*)

Tahap ini merupakan tahapan komunikasi dengan pemangku kepentingan yang akan terlibat dengan sistem. Tahapan komunikasi dapat disebut sebagai proses elisitasi yang merupakan suatu aktivitas berkomunikasi dengan para pemangku kepentingan yang terlibat di dalam pengembangan sistem yang bertujuan untuk mendapatkan kebutuhan sistem (Sommerville dan Sawyer dalam Siahaan, 2012). Teknik yang dapat digunakan dalam proses elisitasi meliputi teknik wawancara, kuisisioner, observasi dan analisis dokumen. Pada penelitian ini teknik yang dilakukan dalam proses elisitasi adalah teknik wawancara untuk mendapatkan gambaran umum sistem dan artifak dokumen dari sistem perizinan.

2. Perencanaan (*Planning*)

Setelah tahap komunikasi proses selanjutnya adalah perencanaan (*Planning*). Prinsip perencanaan pada dasarnya menyediakan panduan untuk mengembangkan *roadmap* yang terbaik bagi perjalanan menuju sistem atau produk yang lengkap (Pressman, 2010). Proses yang dilakukan dalam tahap ini adalah menyusun strategi, teknik dan kualitas untuk mencapai tujuan sistem. Proses ini memberikan gambaran tentang langkah-langkah pengerjaan atau apa yang akan dikerjakan dan melakukan estimasi perubahan.

3. Pemodelan (*Modelling*)

Pada tahap ini proses yang dilakukan adalah menganalisis kebutuhan sistem. Kebutuhan sistem didapatkan dari hasil proses komunikasi untuk dikelompokkan menjadi kebutuhan fungsional dan nonfungsional. Kebutuhan fungsional berisi layanan atau fungsi yang disediakan oleh sistem sedangkan kebutuhan nonfungsional berisi batasan dari sistem. Selain menganalisis kebutuhan, tahap ini juga membuat perancangan sistem seperti perancangan struktur data, arsitektur perangkat lunak dan representasi antarmuka. Proses perancangan menterjemahkan hasil dari analisis kebutuhan ke dalam representasi perangkat lunak. Tujuan dibuat perancangan yaitu untuk mengidentifikasi dan menggambarkan abstraksi dasar sistem beserta dengan relasi-relasinya.

4. Konstruksi (*Construction*)

Proses yang dilakukan pada tahap ini adalah mengubah perancangan sistem ke dalam implementasi kode program. Kode program akan bergantung pada hasil perancangan. Selain melakukan implementasi kode program tahap ini juga melakukan pengujian program. Pengujian program bertujuan untuk melihat program yang dibuat, untuk menemukan kesalahan atau *error* yang disebabkan oleh kesalahan logika, masukan ataupun kesalahan dalam kode. Pressman (2010) berpendapat bahwa pengujian merupakan suatu fungsi kendali yang tujuan utamanya yaitu untuk mendapatkan kesalahan dalam *software* atau program. Pada penelitian ini pengujian dibagi menjadi dua yaitu untuk pengujian kebutuhan fungsional dan kebutuhan nonfungsional. Pengujian kebutuhan fungsional menggunakan pengujian unit, pengujian integrasi dan pengujian validasi.

a. Pengujian Unit

Pengujian unit merupakan pengujian yang berfokus pada upaya verifikasi terhadap unit terkecil dari perancangan perangkat lunak, komponen atau modul perangkat lunak (Pressman, 2010).

b. Pengujian Integrasi

Pengujian integrasi berfokus pada keterkaitan hubungan antar method ataupun antar klas.

c. Pengujian Validasi

Pengujian validasi merupakan pengujian yang berfokus pada aksi-aksi yang terlihat oleh pengguna. Pengujian validasi dikatakan berhasil apabila sistem berfungsi sesuai dengan cara yang diharapkan pengguna (Pressman, 2010).

Pengujian kebutuhan non-fungsional menggunakan pengujian *compatibility*. Pengujian ini bertujuan untuk memeriksa apakah sistem yang dibangun dapat digunakan di berbagai perangkat *browser*. Perbedaan *browser* terkadang memberikan hasil yang sedikit berbeda tanpa memperhatikan tingkat standarisasi HTML dalam aplikasi web (Pressman, 2010).

Selain itu, ada dua metode pengujian yang digunakan yaitu metode pengujian *whitebox* dan pengujian *blackbox*.

1. Pengujian *whitebox* disebut juga pengujian *glassbox* merupakan desain *test case* yang menggunakan struktur kontrol desain prosedural untuk memperoleh *test case*. Dengan menggunakan pengujian *whitebox*, perekraya sistem memperoleh

test case seperti memberikan jaminan bahwa semua jalur independen pada suatu modul telah digunakan paling tidak satu kali, menggunakan semua keputusan logis pada sisi *true* dan *false*, mengeksekusi semua *loop* pada batasan mereka dan pada batasan operasional mereka, dan menggunakan struktur data internal untuk menjamin validasinya (Pressman, 2010). Penerapan pengujian ini digunakan untuk pengujian unit dan pengujian integrasi.

2. Pengujian *blackbox* digunakan untuk pengujian validasi. Pengujian ini memungkinkan untuk membuat beberapa kondisi masukan dari semua kebutuhan fungsional program (Pressman, 2010). Ada banyak jenis metode pengujian *blackbox*, namun untuk penelitian ini digunakan metode pengujian berbasis skenario berdasarkan pada skenario use case yang dibuat. Pengujian berbasis skenario ini berfokus pada apa yang dikerjakan pengguna, bukan apa yang dikerjakan produk (Pressman, 2010).

5. Deployment

Setelah dilakukan implementasi kode program dan pengujian perangkat lunak, maka perangkat lunak yang telah jadi tersebut dikembalikan ke pengguna untuk menguji apakah perangkat lunak yang dikembangkan sudah sesuai dengan maksud dan tujuan pengguna.

2.3 Pemodelan Berorientasi Objek

Pemodelan merupakan sebuah proses merancang perangkat lunak atau sistem sebelum dilakukan implementasi atau *coding* (Yasin, 2012). Pemodelan dalam perangkat lunak digunakan untuk menggambarkan atau memvisualisasikan produk perangkat lunak yang dibuat. Dalam pemodelan perangkat lunak tentu ada perangkat pemodelan, salah satunya yaitu pemodelan *Unified Modelling Language* atau UML. UML sendiri merupakan bahasa pemodelan berorientasi objek. Dalam pengertiannya UML bisa disebut sebagai bahasa visual atau notasi grafis yang didukung oleh meta-model tunggal untuk membantu dalam mendeskripsikan dan merancang perangkat lunak yang dibangun menggunakan pendekatan berorientasi object (Fowler, 2003). UML digunakan untuk menspesifikasikan, memodelkan, membangun, menggambarkan dan mendokumentasikan sistem yang dibuat. Adapun tujuan pemodelan UML adalah (Yasin, 2012):

1. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan sistem yang dapat saling menukar model dengan mudah dan dimengerti secara umum.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.

Pemodelan UML yang digunakan dalam penelitian ini meliputi diagram *use case*, skenario *use case*, diagram *sequence* dan diagram kelas.


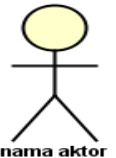


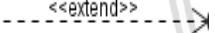
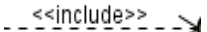
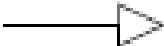
2.3.1 Diagram Use case

Diagram *use case* merupakan suatu pemodelan untuk perilaku dari sistem yang dibuat. *Use case* mendeskripsikan interaksi antara para pengguna sistem (aktor) dengan sistem itu sendiri (Fowler, 2003). *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di

dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Dalam penamaan *use case* lebih baik dibuat simple dan mudah dipahami agar pembaca *use case* dapat memahami interaksi dan alur kerja *use case*.

Simbol-simbol yang digunakan dalam perancangan *use case* diagram terdapat pada Tabel 2.1

Tabel 2.1 Simbol Diagram *Use case*

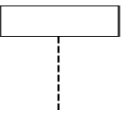
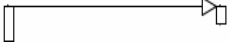


No	Simbol	Nama	Deskripsi
1.		<i>Use case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau actor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
2.		Aktor / <i>actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri; biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i> .
3.		<i>Association</i>	Komunikasi antara aktor dan <i>use case</i> yang saling berinteraksi.
4.		<i>Dependency</i>	Hubungan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).
5.		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.
6.		<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya.
7.		<i>Generalization</i>	Hubungan antara general <i>use case</i> dengan spesifik <i>use case</i> yang saling berbagi perilaku dan struktur data. General <i>use case</i> akan berbagi fungsi atau perilaku untuk spesifik <i>use case</i> .

Sumber: Rumbaugh, et al., (2004)

2.3.2 Diagram *Sequence*

Diagram *sequence* digunakan untuk memodelkan interaksi antara aktor dan objek dalam suatu sistem dan interaksi antara objek itu sendiri (Sommerville, 2011). Diagram tersebut mendeskripsikan objek dan pesan yang dilewatkan diantara objek-objek di dalam *use case* (Fowler, 2003). Untuk menggambarkan diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram *sequence* juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Berikut merupakan simbol-simbol yang ada pada diagram *sequence* terdapat pada Gambar 2.2

Tabel 2.2 Simbol Diagram *Sequence*

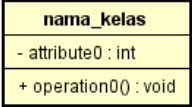

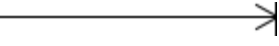
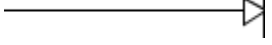
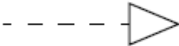
No	Simbol	Nama	Deskripsi
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
4		<i>Return</i>	Berfungsi untuk menyatakan nilai kembalian dari sebuah operasi yang dijalankan pada suatu objek.

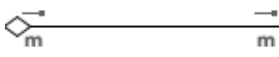
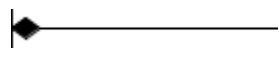
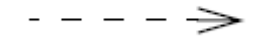
Sumber: Rumbaugh, et al., (2004)

2.3.3 Diagram Klas

Diagram klas menggambarkan jenis objek dalam sistem dan berbagai jenis relasi static yang ada (Fowler, 2003). Diagram klas juga menampilkan atribut dan operasi dari kelas dan relasi dari objek yang saling terhubung. Klas adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek. Klas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu klas. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu klas. Adapun simbol-simbol diagram klas ditunjukkan pada Tabel 2.3

Tabel 2.3 Simbol Diagram Klas

No	Simbol	Nama	Deskripsi
1.		Klas	Klas pada struktur sistem.
2.		<i>Association</i>	Relasi antara <i>instance</i> klas.
3.		<i>Directed association</i>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4.		Generalisasi	Relasi antara deskripsi yang lebih umum dan variasi yang lebih spesifik (umum-khusus) yang digunakan sebagai pewarisan atau <i>inheritance</i> .
5.		<i>realization</i>	Relasi antara spesifikasi dan implementasinya.

6.		<i>agregation</i>	Relasi asosiasi yang mewakili keseluruhan hubungan (<i>part-whole</i>)
7.		<i>composition</i>	Bentuk relasi asosiasi yang lebih kuat dimana komposit memiliki tanggung jawab penuh untuk mengelola bagian-bagiannya. Seperti <i>allocation</i> dan <i>deallocation</i>
8.		<i>depedency</i>	Relasi antara dua elemen model.

Sumber: Rumbaugh, et al., (2004)

2.4 Teknologi Pengembangan Sistem

Teknologi yang digunakan dalam pembangunan sistem perizinan adalah sebagai berikut:

2.4.1 PHP

PHP merupakan kepanjangan dari *Hypertext Preprocessor*. PHP merupakan bahasa pemrograman yang berjalan pada server yang hasilnya dapat ditampilkan pada *client* (Nugroho, 2004). PHP diciptakan oleh Rasmus Lerdorf yang membuat script perl. Pada tahun 1994, Rasmus Lerdorf membuat script Perl dan membuat resumennya untuk menarik perhatian pengguna atau programmer. Ternyata hasilnya menakjubkan dan banyak programmer yang menyukai *script* tersebut sehingga dibuatlah package yaitu Personal Home Page. Dari situlah diciptakanlah tool lain untuk mengambil input dari HTML. FormFI atau Form Interpreter juga dikenali sebagai PHP/FI atau PHP 2. Banyak pengguna yang tertarik untuk mengerjakan proyek ini dan dikembangkan lagi menjadi PHP3. PHP3 terus dikembangkan dengan menambah simple API. Seiring berjalannya waktu PHP terus berkembang sampai saat ini yang sudah bisa digunakan dengan kolaborasi database (Nugroho, 2004). Adapun sistem *database* yang telah didukung oleh PHP adalah *Oracle*, *Sysbase*, *mSQL*, *MySQL* dan *Solid* (Nugroho, 2004).

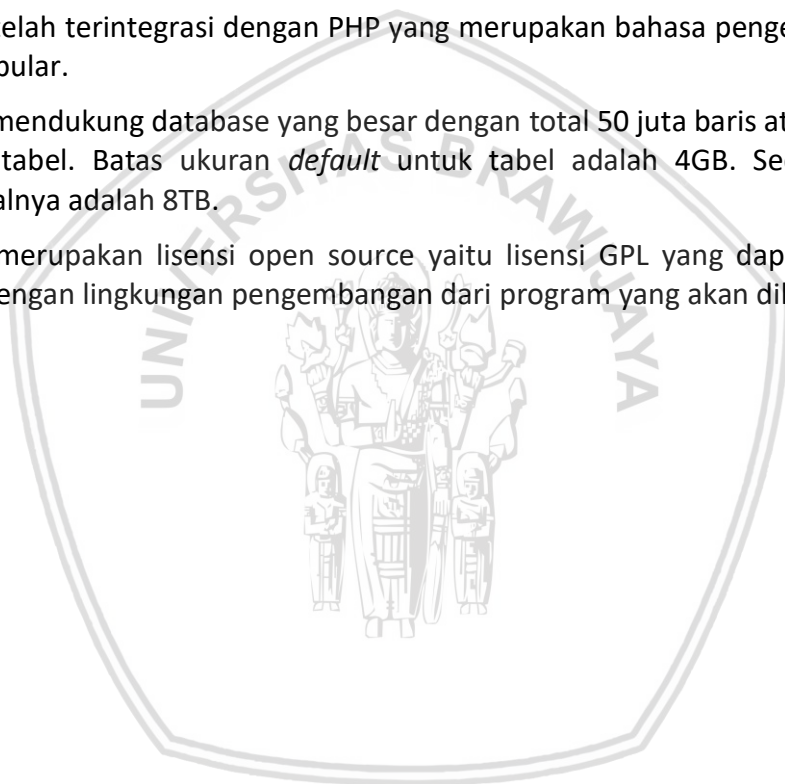
PHP merupakan *software* yang bersifat *open source* dan mampu lintas platform, artinya PHP dapat dijalankan serta digunakan pada sistem operasi dan *website server* apapun. PHP mampu berjalan di *windows* maupun di *linux*. PHP dapat dijalankan melalui dua cara yaitu sebagai modul di *Apache website server* dan sebagai *binary* di CGI. Fungsi utama PHP yaitu untuk menghubungkan *database* dengan web. Adanya PHP ini membuat aplikasi berbasis web dapat terkoneksi dengan mudah ke *database*.

2.4.2 MySQL

MySQL dikenal sebagai open source SQL yang paling populer. MySQL merupakan sistem manajemen basisdata yang dapat dikembangkan, didistribusikan dan didukung oleh Oracle Corporation (Axmark & Widenius, 1997). MySQL merupakan database relational yang bersifat open source yang bisa digunakan atau dimodifikasi oleh semua kalangan tanpa membayar. MySQL disebut sebagai *database server* yang cepat, handal, *scalable* dan mudah digunakan yang dapat bekerja di *client* atau *server* atau *embedded system* (Axmark & Widenius, 1997). Sebagian besar perangkat lunak juga mendukung MySQL software.

MySQL dibuat oleh TcX dan telah dipercaya mengelola system dengan 40 buah database berisi 10.000 tabel dan 500 di antaranya memiliki 71 juta baris. MySQL dapat digunakan secara berdiri sendiri maupun di lekatkan pada bahasa pemrograman seperti C, dan Delphi. Kelebihan MySQL yaitu (Tutorialspoint, 2016):

1. MySQL dirilis dibawah lisensi *open source*.
2. MySQL adalah program yang *powerfull* yang dapat menangani fungsionalitas dari *database* yang *powerfull* dan mahal.
3. MySQL menggunakan bahasa standar data SQL.
4. MySQL bekerja di berbagai jenis sistem operasi dan didukung banyak bahasa pemrograman seperti PHP, PERL, C, C++, JAVA.
5. MySQL dapat bekerja dengan baik dan cepat dalam data yang besar.
6. MySQL telah terintegrasi dengan PHP yang merupakan bahasa pengembangan web yang populer.
7. MySQL mendukung database yang besar dengan total 50 juta baris atau lebih dalam sebuah tabel. Batas ukuran *default* untuk tabel adalah 4GB. Sedangkan batas maksimalnya adalah 8TB.
8. MySQL merupakan lisensi open source yaitu lisensi GPL yang dapat dimodifikasi sesuai dengan lingkungan pengembangan dari program yang akan dikembangkan.

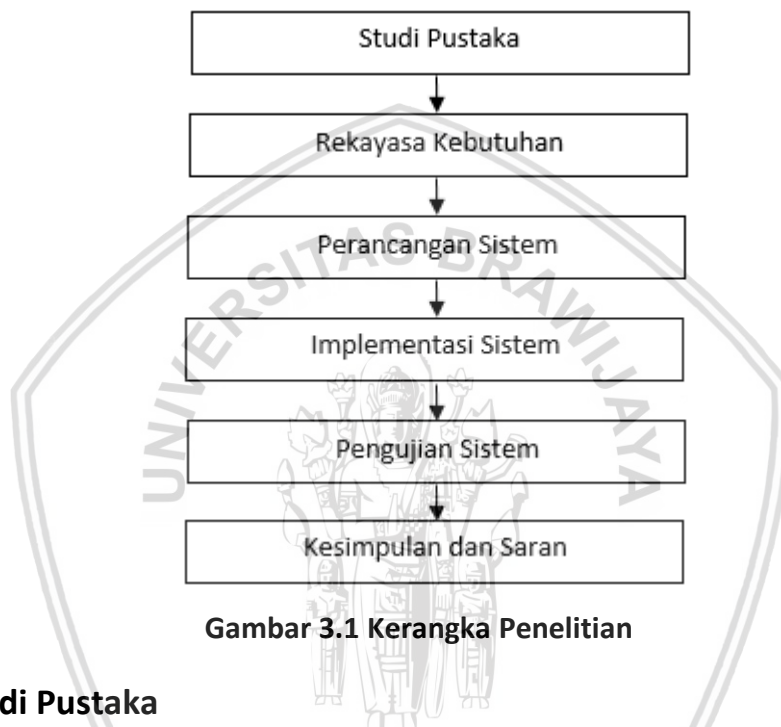






BAB 3 METODOLOGI

Dalam Bab ini menjabarkan tahapan yang dilakukan peneliti untuk pembangunan sistem perizinan. Tahapan tersebut digambarkan dalam diagram alur penelitian yang ditunjukkan pada Gambar 3.1. Tahapan pertama yang dilakukan adalah melakukan pengumpulan data, mencari referensi, literatur yang berhubungan dengan penelitian ini kemudian melakukan analisis kebutuhan sistem, perancangan sistem, implementasi, pengujian, mendapatkan kesimpulan dan saran.



Gambar 3.1 Kerangka Penelitian

3.1 Studi Pustaka

Studi pustaka merupakan pencarian literatur dan referensi yang digunakan untuk mendukung penulisan penelitian. Studi pustaka dapat juga digunakan untuk membandingkan hasil penelitian sebelumnya. Pada penelitian ini studi pustaka digunakan sebagai referensi untuk menunjang topik penelitian tentang pembangunan sistem perizinan. Referensi yang digunakan diambil dari berbagai media seperti buku, e-book, jurnal dan website. Referensi yang berhubungan dengan penelitian ini meliputi:

1. Pengertian dan fungsi perizinan.
2. Metode Pengembangan Sistem.
3. Diagram UML (Unified Modeling Language).
4. Pemodelan Berorientasi Objek.
5. Teknologi pengembangan sistem yang meliputi
 - PHP
 - MySQL

3.2 Rekayasa Kebutuhan

Setelah melakukan studi pustaka maka tahap selanjutnya adalah tahap rekayasa kebutuhan. Tahap ini digunakan untuk mendapatkan kebutuhan sistem perizinan. Tahap ini dimulai dari proses elisitasi kebutuhan, identifikasi aktor, membuat daftar kebutuhan dan pemodelan kebutuhan.

1. Elisitasi kebutuhan dilakukan dengan teknik wawancara. Hal-hal yang digali dari tahap ini yaitu proses bisnis sistem perizinan yang akan dikembangkan dan kebutuhan sistem yang akan digunakan menjadi fitur sistem. Wawancara dilakukan secara langsung dengan kepala seksi pengembangan dan pengelolaan aplikasi pada Dinas Kominfo Kota Pasuruan. Wawancara dilakukan dengan berfokus pada:
 - a. Permasalahan yang sedang terjadi dan mendapatkan gambaran umum sistem yang dibuat sebagai solusi dalam mengatasi permasalahan tersebut.
 - b. Mengidentifikasi siapa saja pemangku kepentingan yang akan terlibat di dalam sistem.
 - c. Mengumpulkan artifak dokumen mengenai sistem yang akan dibangun.

Dari hasil elisitasi didapatkan kebutuhan yang diinginkan oleh pemangku kepentingan yang dapat dijadikan acuan dalam kebutuhan sistem. Digambarkan juga *Work Breakdown Structure* (WBS) dari sistem perizinan yang menjabarkan fungsi-fungsi yang terdapat dalam sistem. Selain itu, digambarkan juga proses bisnis perizinan.

2. Setelah proses elisitasi akan dilakukan identifikasi aktor dan mendeskripsikan peran dari setiap aktor yang terlibat.
3. Selanjutnya membuat daftar kebutuhan, kebutuhan akan dikelompokkan menjadi kebutuhan fungsional dan non-fungsional. Pengelompokan kebutuhan digunakan untuk menentukan prioritas kebutuhan yang akan digunakan dalam sistem perizinan.
4. Setelah mendapatkan daftar kebutuhan fungsional, maka dilakukan pemodelan kebutuhan untuk menggambarkan interaksi antar sistem dengan aktor yang dibuat dalam bentuk diagram *use case* dan dilanjutkan dengan skenario *use case* untuk menggambarkan alur kerja dari setiap *use case*.

3.3 Perancangan Sistem

Perancangan sistem merupakan tahapan yang dilakukan setelah proses rekayasa kebutuhan. Tujuan dibuat perancangan yaitu untuk menentukan relasi antar komponen, mekanisme komunikasi setiap komponen dan antarmuka komponen. Selain itu perancangan dibuat untuk memberikan gambaran yang jelas terhadap sistem perizinan yang dibangun. Tahap ini mengacu dari kebutuhan

sistem yang didapat pada tahap rekayasa kebutuhan. Perancangan yang dilakukan pada penelitian ini adalah perancangan arsitektur, pemodelan diagram *sequence*, pemodelan diagram klas, perancangan data dan perancangan antarmuka.

1. Perancangan arsitektur memberikan gambaran mengenai komponen fungsional perangkat lunak yang disusun, diorganisasi dan distrukturkan sehingga hubungan antar komponen dapat dijelaskan, antarmuka dapat didefinisikan dalam sistem.
2. Pemodelan diagram *sequence* adalah perancangan yang menggambarkan perilaku objek pada *use case* dengan mendefinisikan waktu hidup objek dan pesan yang akan dikirimkan dan diterima antar objek. Pemodelan diagram *sequence* dibuat berdasarkan dari skenario *use case* yang telah dibuat pada tahap analisis kebutuhan.
3. Pemodelan diagram klas menggambarkan klas-klas yang ada pada sistem dan menunjukkan bagaimana relasi setiap klasnya.
4. Perancangan data akan menggambarkan struktur data yang digunakan pada implementasi sistem yang dibangun. Tahapan pada perancangan data dimulai dari menganalisis kebutuhan basis data berdasarkan dari kebutuhan sistem. Perancangan konseptual basis data akan digambarkan menggunakan diagram ERD (*Entity Relationship Diagram*) untuk menentukan relasi antar entitas di dalam sistem. Selanjutnya penjabaran dari ERD yaitu menggambarkan desain *logic* dari sistem menggunakan CDM (*Conceptual Data Model*) dan terakhir merancang desain fisik basis data menggunakan PDM (*Physical Data Model*) yang nantinya dijadikan acuan dalam implementasi basis data ke dalam DBMS.
5. Perancangan antarmuka memberikan gambaran desain atau tampilan sistem yang akan dibuat.

3.4 Implementasi Sistem

Implementasi sistem merupakan tahapan yang dilakukan setelah perancangan sistem. Implementasi akan mengubah spesifikasi sistem menjadi sistem yang dapat dijalankan. Pembangunan sistem akan dikembangkan dengan berbasis website menggunakan bahasa pendukung PHP (*Hypertext Preprocessor*) dan MySQL sebagai penyimpanan dan manajemen basis data. Dalam implementasi sistem dijelaskan mengenai spesifikasi lingkungan pengembangan, implementasi basis data, implementasi klas dan asset pada file program, dan implementasi antarmuka.

3.5 Pengujian Sistem

Pengujian sistem digunakan untuk melihat adanya *bug* atau *error* yang terdapat pada sistem. Pengujian dilakukan untuk memastikan bahwa sistem yang dibuat sudah sesuai dengan perancangan yang telah didefinisikan. Dalam penelitian ini pengujian yang digunakan adalah pengujian *whitebox* dan *blackbox*.

Pengujian *whitebox* digunakan untuk pengujian integrasi. Pengujian *blackbox* digunakan untuk pengujian unit dan pengujian validasi.

1. Pengujian unit merupakan pengujian dengan menguji unit terkecil dari sistem untuk mengetahui apakah unit sistem tersebut dapat digunakan atau tidak. Dalam penelitian ini pengujian unit dilakukan menggunakan teknik *basis path testing*. *Basis path testing* digunakan untuk menguji kode program berdasarkan algoritme pada setiap metode yang ada di klas.
2. Pengujian integrasi digunakan untuk menguji dua unit atau lebih yang saling berhubungan. Metode yang digunakan yaitu pengujian *whitebox*. Teknik yang digunakan dalam penelitian ini adalah *basis path testing*. *Basis path testing* untuk menguji kode program berdasarkan algoritme pada setiap metode yang ada di klas.
3. Pengujian validasi adalah pengujian yang dilakukan untuk mengetahui apakah sistem yang dibangun sudah sesuai dengan daftar kebutuhan yang telah didefinisikan sebelumnya. Item-item yang telah dirumuskan dalam daftar kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Metode yang digunakan dalam pengujian validasi untuk sistem ini adalah pengujian *blackbox*.

3.6 Kesimpulan dan Saran

Kesimpulan disebut sebagai pernyataan singkat mengenai keseluruhan hasil dari penelitian yang telah dibahas pada Bab sebelumnya. Kesimpulan dapat menjawab pertanyaan yang diajukan pada rumusan masalah pada Bab pendahuluan. Sedangkan saran berisi solusi yang ditawarkan untuk memperbaiki dan mengevaluasi apabila terjadi kekurangan pada penelitian ini dan diharapkan dapat dijadikan bahan pertimbangan untuk penelitian selanjutnya.







BAB 4 REKAYASA KEBUTUHAN

Proses rekayasa kebutuhan dapat disebut sebagai aktivitas untuk mencari, mengidentifikasi dan menganalisis kebutuhan sistem perizinan. Dalam Bab ini proses analisis kebutuhan akan dimulai dari gambaran umum sistem, proses rekayasa kebutuhan dan pemodelan kebutuhan.

4.1 Gambaran Umum Sistem

Sistem perizinan merupakan sistem yang digunakan untuk mengajukan permohonan perizinan:

1. SIUP
2. TDP
3. TDUP
4. Reklame

Dalam sistem ini ada tujuh aktor yang terdiri dari pengguna, pemohon, staff *back office*, kepala pengendalian, kepala bidang, kepala dinas dan kepala administrasi. Pemohon dapat menggunakan sistem ini untuk mengajukan izin. Selain pemohon dapat mengajukan izin, sistem ini dapat melihat status pengajuan izin dan kapan bukti perizinan dapat diambil. Pemohon dapat mengambil Surat Keputusan (SK) setelah mendapatkan notifikasi bahwa SK telah selesai dibuat. Sedangkan *back office* mempunyai wewenang untuk memeriksa berkas perizinan yang diajukan pemohon, membuat SK, memvalidasi berkas perizinan dan menerbitkan SK perizinan. Selain itu *back office* juga dapat melihat daftar pemohon yang sedang mengajukan izin dan menghapus pemohon. Pada Gambar 4.1 menggambarkan keseluruhan fungsi-fungsi yang terdapat dalam sistem perizinan dengan menggunakan *Work Breakdown Structure* (WBS).

4.2 Proses Rekayasa Kebutuhan

Proses rekayasa kebutuhan merupakan langkah-langkah yang digunakan dalam menggali kebutuhan dari para pemangku kepentingan untuk mendapatkan kebutuhan sistem yang tepat.

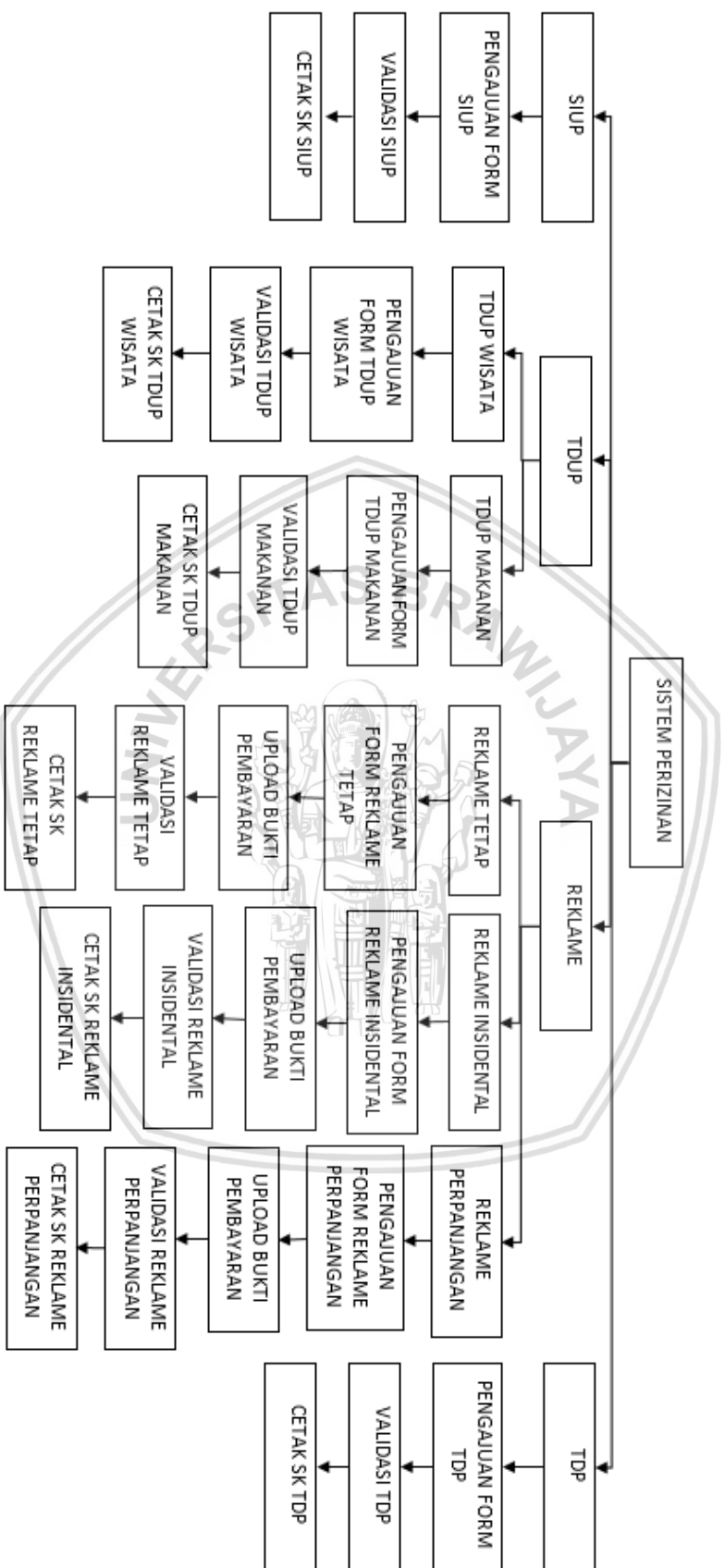
4.2.1 Elisitasi Kebutuhan

Elisitasi adalah tahapan penggalan kebutuhan. Tahap ini bertujuan untuk membantu pengembang dalam memahami sistem yang akan dispesifikasi. Dari hasil elisitasi dengan pemangku kepentingan didapatkan beberapa kebutuhan sistem yang dijabarkan pada Tabel 4.1

Tabel 4.1 Kebutuhan Hasil Wawancara

No	Kebutuhan Yang Didapat
1.	Pemohon yang dapat mengajukan izin adalah pemohon yang telah terdaftar di sistem.
2.	Pemohon dapat melihat proses pengajuan perizinan sudah sampai tahap mana.
3.	Setiap perizinan memiliki persyaratan yang berbeda-beda. Untuk perizinan SIUP, TDP dan TDUP mempunyai alur yang sama sedangkan untuk perizinan reklame prosesnya berbeda yaitu berkas akan divalidasi ketika pemohon sudah mengupload bukti pembayaran.
1.	Berkas yang masuk akan ditanggapi langsung oleh staff <i>back office</i> untuk memeriksa kelengkapan berkas perizinan.
5.	Kepala administrasi hanya bertugas untuk pemberian nomor surat.
6.	Kepala pengendalian bertugas memeriksa ulang kelengkapan berkas serta menyetujui atau menolak berkas pemohon yang telah divalidasi oleh staff <i>back office</i> .
7.	Kepala bidang pelayanan terpadu satu pintu mempunyai tugas untuk menvalidasi draft atau berkas pemohon.
8.	Kepala dinas perizinan memiliki kewenangan untuk menyetujui atau menolak persetujuan izin. Kepala dinas juga dapat melihat siapa saja pemohon yang telah mengajukan izin. Setiap perizinan yang masuk kebagian kepala dinas telah dibuat SK perizinannya.
9.	Jika berkas telah disetujui oleh kepala dinas, berkas akan dicetak oleh staff <i>back office</i> .
10.	Sistem ini dapat diimplementasikan pada browser dengan jenis <i>browser Google Chrome, Mozilla Firefox dan Internet Explorer</i> .

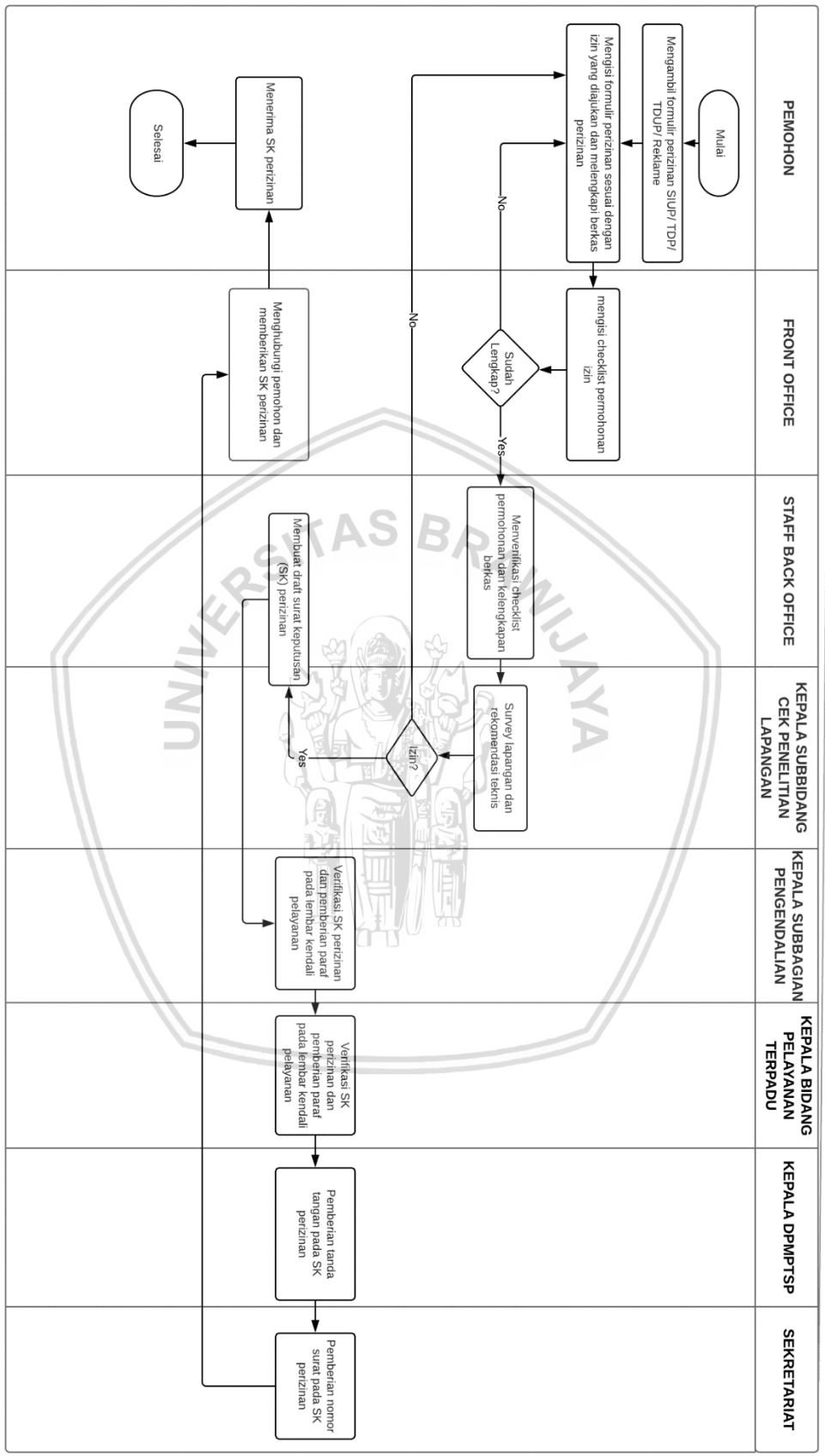
Selain mendapatkan kebutuhan sistem, proses elisitasi yang dilakukan juga menggambarkan proses bisnis perizinan yang sedang digunakan dan proses bisnis perizinan yang akan dibuat dalam sistem. Proses bisnis yang sedang digunakan dapat dilihat pada Gambar 4.2 dan proses bisnis yang akan dibuat dalam sistem dapat dilihat pada Gambar 4.3.



Gambar 4.1 Work Breakdown Structure

PROSES PERIZINAN YANG DIGUNAKAN SAAT INI

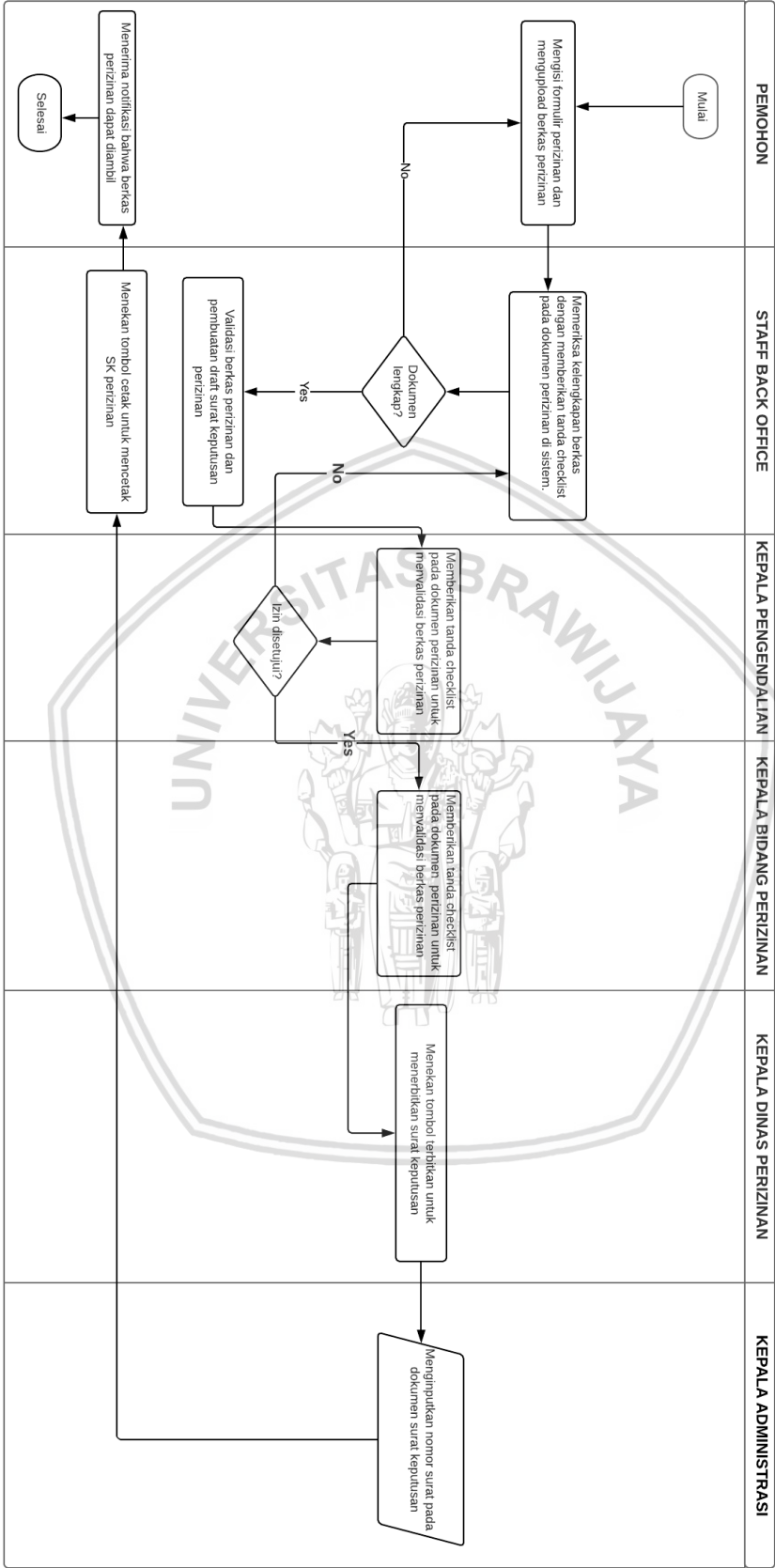
Anisah Denis |



Gambar 4.2 Proses Bisnis Perizinan Yang digunakan

PROSES PERIZINAN SISTEM

Anisah Denis |



Gambar 4.3 Proses Bisnis Perizinan Sistem Yang Baru

Pada Gambar 4.2 Proses perizinan dimulai dengan pemohon yang mengajukan perizinan melalui bagian *front office*. Bagian *front office* akan memberikan pemohon formulir perizinan sesuai dengan izin yang diajukan. Kemudian pemohon mengisi formulir dan melengkapi berkas yang menjadi persyaratan dalam mengajukan perizinan. Setelah selesai mengisi formulir dan melengkapi berkas, pemohon kemudian menyerahkan formulir tersebut ke bagian *front office*. Bagian *front office* ini yang menyerahkan permohonan perizinan ke bagian staff *back office*. Staff *back office* menerima pengajuan permohonan izin, memeriksa berkas pemohon dan membuat SK perizinan dan *form* kontrol. Staff *back office* inilah yang akan menemui kasubid lapangan, ketua bidang dan ketua DPMPTSP untuk memvalidasi berkas dan memberikan paraf. Setelah disetujui proses perizinannya, maka staff *back office* memberikan SK perizinan kepada bagian *front office* dan bagian *front office* menghubungi pemohon bahwa SK perizinan dapat diambil.

Pada Gambar 4.3 merupakan proses bisnis perizinan sistem. Proses dimulai dengan pemohon memilih menu dari izin yang akan diajukan. Pemohon mengisi formulir perizinan dan mengupload berkas perizinan. Berkas perizinan yang diupload pemohon akan diperiksa oleh staff *back office*. Staff *back office* ini akan memeriksa kelengkapan berkas dan formulir serta membuat SK perizinan. Staff *back office* akan mengirimkan berkas perizinan pemohon kepada kepala pengendalian dan kepala bidang untuk memvalidasi berkas pemohon. Setelah itu berkas akan dikirim ke bagian kepala dinas untuk menerbitkan SK perizinan yang telah dibuat oleh staff *back office*. SK perizinan tadi akan diberikan nomor surat oleh kepala administrasi dan akan dicetak oleh bagian *back office* sebelum diberikan kepada pemohon.

4.2.2 Identifikasi Aktor

Dari hasil elisitasi kebutuhan, didapatkan pemangku kepentingan yang dapat dijadikan aktor yang terlibat di dalam sistem. Aktor yang terlibat adalah pengguna pemohon, staff *back office*, kepala administrasi, kepala pengendalian, kepala bidang pelayanan terpadu satu pintu dan kepala dinas perizinan. Pada Tabel 4.2 akan dijelaskan mengenai aktor yang dapat berperan dalam sistem dan deskripsi setiap aktor.

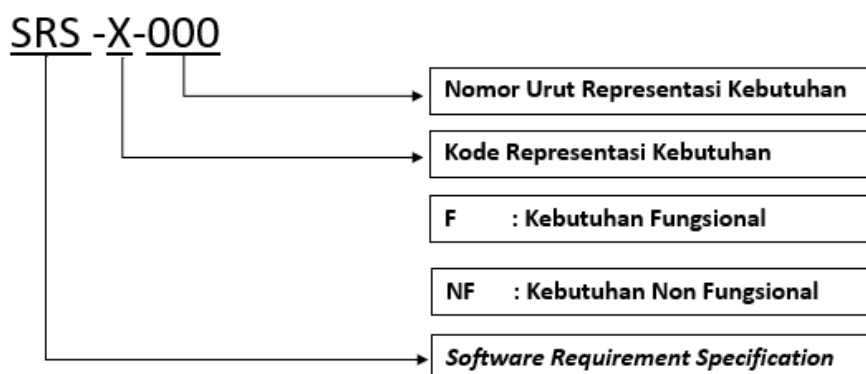
Tabel 4.2 Identifikasi Aktor

Aktor	Deskripsi
Pengguna	Pengguna merupakan aktor sistem yang belum terdaftar. Aktor ini hanya dapat melakukan registrasi dan login
Pemohon	Pemohon merupakan aktor yang sudah terdaftar disistem. Pemohon dapat memperbarui biodata, mengajukan perizinan SIUP, TDP, TDUP dan Reklame,

	menerima notifikasi perizinan, melihat status perizinan.
Staff Back office	Staff <i>back office</i> merupakan aktor yang telah terdaftar di sistem. Aktor ini dapat memeriksa kelengkapan berkas dan memvalidasi berkas pemohon, membuat SK perizinan.
Kepala Administrasi	Kepala administrasi merupakan aktor yang dapat memberikan nomor surat.
Kepala Pengendalian	Kepala pengendalian merupakan aktor yang mempunyai tugas untuk memeriksa dan menyetujui berkas pemohon. Selain itu kepala pengendalian dapat juga melihat daftar pemohon dan menghapus akun pemohon.
Kepala Bidang	Kepala bidang merupakan aktor yang mempunyai tugas untuk memeriksa berkas pemohon dan menyetujui untuk berkas pemohon. Selain itu kepala bidang dapat melihat daftar pemohon.
Kepala Dinas	Kepala dinas perizinan merupakan aktor yang mempunyai tugas untuk menerbitkan SK perizinan. Kepala dinas juga dapat melihat pemohon yang sedang mengajukan perizinan.

4.2.3 Daftar Kebutuhan

Berdasarkan dari elisitasi kebutuhan dan pemodelan proses bisnis yang dijelaskan sebelumnya, kebutuhan akan dianalisis dan dikelompokkan menjadi kebutuhan fungsional dan kebutuhan nonfungsional. Untuk memudahkan pengelompokkan, maka didefinisikan terlebih dahulu aturan penomoran kebutuhan. Aturan penomoran tersebut dapat dilihat pada Gambar 4.4



Gambar 4.4 Aturan Penomoran Spesifikasi Kebutuhan

4.2.3.1 Daftar Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang mendeskripsikan layanan, fungsi pada suatu sistem bagi penggunanya. Pada Tabel 4.3 menjelaskan daftar kebutuhan yang didapat dari proses elisitasi kebutuhan.

Tabel 4.3 Daftar Kebutuhan Fungsional

No	Kode Spesifikasi	Daftar Kebutuhan	Aktor
1.	SRS-F-100	Sistem harus menyediakan layanan registrasi pengguna sehingga pengguna dapat menggunakan fungsionalitas sistem.	Pengguna
1.1	SRS-F-101	Pengguna harus disediakan formulir registrasi untuk mendaftar ke sistem dengan menggunakan NIK minimal 16 karakter.	
2.	SRS-F-200	Sistem dapat menyediakan layanan <i>login</i> untuk pengguna masuk ke sistem	Pemohon
2.1	SRS-F-201	Pengguna harus disediakan formulir <i>login</i> agar dapat masuk ke sistem menggunakan <i>username</i> dan <i>password</i> .	
3.	SRS-F-300	Sistem dapat menyediakan layanan <i>login</i> untuk staff <i>back office</i> , kepala pengendalian, kepala bidang, kepala administrasi, kepala dinas masuk ke sistem	Staff <i>back office</i> , kepala pengendalian, kepala bidang, kepala administrasi, kepala dinas
3.1	SRS-F-301	Pengguna harus disediakan formulir <i>login</i> agar dapat masuk ke sistem menggunakan <i>username</i> , <i>password</i> dan level.	Staff <i>back office</i> , kepala pengendalian, kepala bidang, kepala administrasi, kepala dinas

4.	SRS-F-400	Sistem dapat menyediakan layanan <i>logout</i> untuk pengguna keluar dari sistem	Pemohon, <i>staff back office</i> , kepala pengendalian, kepala bidang, kepala administrasi, kepala dinas
5.	SRS-F-500	Sistem harus menyediakan layanan untuk mengedit biodata pemohon.	Pemohon
5.1	SRS-F-501	Sistem harus menyediakan formulir untuk pemohon mengedit biodata yang berisi identitas diri pemohon seperti nama lengkap pemohon, jenis kelamin, tempat dan tanggal lahir serta pekerjaan.	
6.	SRS-F-600	Sistem dapat menampilkan pemberitahuan.	Pemohon
7.	SRS-F-700	Sistem harus menyediakan layanan untuk pemohon mengajukan permohonan izin SIUP	Pemohon
7.1	SRS-F-701	Sistem harus menyediakan formulir permohonan izin SIUP yang berisi identitas perusahaan, akta pendirian, akta perubahan dan upload berkas persyaratan pemohon dengan tipe file jpg atau png.	
8.	SRS-F-800	Sistem menyediakan layanan kepada pemohon agar dapat mengajukan permohonan izin TDP	Pemohon
8.1	SRS-F-801	Sistem harus menyediakan formulir permohonan izin TDP yang berisi jenis pendaftaran, identitas perusahaan, akta pendirian, akta perubahan dan upload berkas persyaratan pemohon dengan tipe file jpg atau png.	
9.	SRS-F-900	Sistem menyediakan layanan kepada pemohon agar dapat mengajukan permohonan izin TDUP	Pemohon

9.1	SRS-F-901	Sistem harus menyediakan formulir permohonan izin TDUP yang identitas perusahaan, identitas pemohon dan upload berkas persyaratan pemohon dengan tipe file jpg atau png.	
10.	SRS-F-1000	Sistem menyediakan layanan kepada pemohon agar dapat mengajukan permohonan izin reklame	Pemohon
10.1	SRS-F-1001	Sistem harus menyediakan formulir permohonan izin Reklame yang berisi identitas perusahaan, identitas pemohon, identitas pemasangan reklame dan upload berkas persyaratan pemohon dengan tipe file jpg atau png.	
11.	SRS-F-1100	Sistem menyediakan layanan kepada pemohon agar pemohon dapat melihat sampai dimana proses permohonan izin yang diajukan.	Pemohon
12.	SRS-F-1200	Sistem menyediakan layanan untuk memeriksa kelengkapan berkas perizinan baik itu perizinan SIUP, TDP, TDUP dan reklame kepada staff <i>back office</i> .	Staff <i>back office</i>
12.1	SRS-F-1201	Sistem menyediakan formulir <i>check list</i> untuk memeriksa berkas perizinan yang diajukan pemohon.	
13.	SRS-F-1300	Sistem menyediakan layanan untuk staff <i>back office</i> membuat surat keputusan perizinan.	Staff <i>back office</i>
13.1	SRS-F-1301	Sistem menyediakan formulir surat keputusan perizinan SIUP/ TDP/ TDUP/ Reklame sesuai dengan yang diajukan pemohon.	
14.	SRS-F-1400	Sistem akan menampilkan cetak surat keputusan (SK) perizinan apabila SK perizinan telah diterbitkan oleh kepala dinas.	Staff <i>back office</i>
15.	SRS-F-1500	Sistem menyediakan layanan kepada kepala seksi administrasi untuk memberikan nomor surat pada SK	Kepala administrasi

		perizinan SIUP, TDP, TDUP dan reklame yang dibuat oleh <i>back office</i> .	
16.	SRS-F-1600	Sistem harus menyediakan layanan agar kepala pengendalian dan kepala bidang dapat memvalidasi berkas perizinan SIUP, TDP, TDUP dan reklame.	Kepala pengendalian, kepala bidang
17.	SRS-F-1700	Sistem menyediakan layanan untuk menerbitkan SK perizinan.	Kepala dinas
18.	SRS-F-1800	Sistem dapat menampilkan daftar pemohon. Daftar pemohon yang ditampilkan berupa nama pemohon, <i>username</i> pemohon dan NIK.	Kepala pengendalian, kepala bidang
19.	SRS-F-1900	Sistem menyediakan layanan untuk menghapus data pemohon.	Kepala pengendalian

4.2.3.2 Daftar Kebutuhan NonFungsional

Kebutuhan nonfungsional adalah kebutuhan yang menggambarkan karakteristik, batasan pada sistem baik untuk lingkungan pengembangan maupun operasional serta mendeskripsikan atribut kualitas yang harus dipenuhi oleh sistem. Kebutuhan nonfungsional pada sistem ini didapatkan berdasarkan hasil dari wawancara dengan kepala pengembangan aplikasi di Dinas Kominfo sehingga menghasilkan kebutuhan nonfungsional yang ditunjukkan pada Tabel 4.4.

Tabel 4.4 Daftar Kebutuhan NonFungsional

No Kebutuhan	Kode Spesifikasi	Daftar kebutuhan
19	SRS-NF-100	Sistem dapat dijalankan di berbagai <i>browser</i> seperti <i>Mozilla Firefox</i> , <i>Google Chrome</i> dan <i>Internet Explorer</i> .

4.3 Pemodelan Kebutuhan

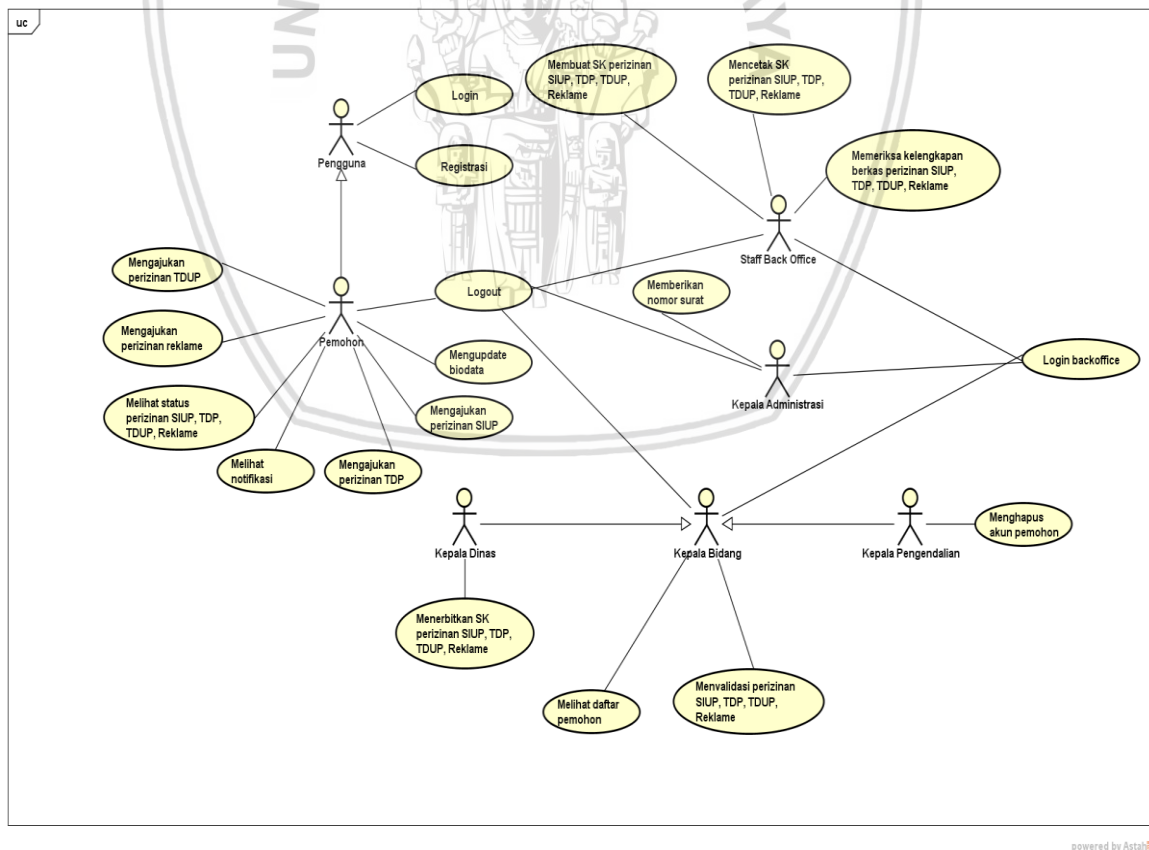
Pemodelan disebut sebagai gambaran dari sistem yang dituangkan dalam diagram sesuai dengan aturan dan prinsip tertentu. Pada penelitian ini pemodelan kebutuhan akan diimplementasikan dalam diagram *use case* dan skenario *use case*.

4.3.1 Diagram *Use case*

Diagram *use case* adalah diagram yang menggambarkan perilaku dan seluruh kebutuhan sistem. Diagram *use case* dibuat untuk mengetahui aktor yang

berperan dalam sistem dan fungsi-fungsi yang ada di dalam sistem. Diagram *use case* digambarkan sesuai dengan identifikasi kebutuhan fungsional. Pada penelitian ini, diagram *use case* ditunjukkan pada Gambar 4.5.

Pada Gambar 4.5 diagram *use case* terdiri dari tujuh aktor yaitu pengguna, pemohon, staff *back office*, kepala pengendalian, kepala bidang, kepala dinas dan kepala administrasi. *Use case login* dan registrasi bisa dilakukan oleh aktor pengguna. *Use case logout*, mengupdate biodata, mengajukan perizinan SIUP, mengajukan perizinan TDP, mengajukan perizinan TDUP, mengajukan perizinan reklame, melihat status perizinan SIUP, TDP, TDUP dan reklame dan melihat notifikasi dapat dilakukan oleh pemohon. Untuk staff *back office* dapat melakukan *login backoffice*, *logout*, proses memeriksa kelengkapan berkas perizinan SIUP, TDP, TDUP dan Reklame, membuat dan mencetak SK perizinan SIUP, TDP, TDUP dan Reklame. Aktor Kepala Pengendalian dapat melakukan *login backoffice*, *logout*, proses validasi berkas perizinan SIUP, TDP, TDUP dan Reklame, melihat dan menghapus daftar pemohon. Kepala Bidang dapat melakukan *login backoffice*, *logout*, proses validasi berkas perizinan SIUP, TDP, TDUP dan Reklame serta melihat daftar pemohon. *Use case login backoffice*, *logout*, memberikan nomor surat dapat dilakukan oleh aktor Kepala Administrasi. *Use case login backoffice*, *logout*, menerbitkan SK perizinan SIUP, TDP, TDUP dan Reklame dapat dilakukan oleh Kepala Dinas.



Gambar 4.5 Diagram *Use Case* Sistem Perizinan

4.3.2 Skenario *Use Case*

Skenario *use case* merupakan urutan alur dari diagram *use case* yang telah ditunjukkan pada Gambar 4.5. Skenario *use case* akan dibuat menggunakan tabel yang terdiri dari *actor*, *objective*, *pre-condition*, *main flow*, *alternative flow*, *post condition*.

Tabel 4.5 Skenario *Use Case* Registrasi

<i>SRS-F-100</i>	Registrasi
<i>Objective</i>	Agar pengguna dapat melakukan registrasi di sistem.
<i>Actor</i>	Pengguna
<i>Pre Condition</i>	Halaman formulir registrasi telah terbuka dan tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. <i>Actor</i> mengisi data pada formulir registrasi berupa <i>username</i>, <i>password</i>, <i>confirm password</i>, NIK dan email. Kemudian <i>actor</i> menekan tombol daftar. 2. Sistem akan melakukan pengecekan terhadap data yang telah diinputkan <i>actor</i> dan data akan disimpan dalam <i>database</i>.
<i>Alternative Flow</i>	Apabila data yang diinputkan <i>actor</i> tidak lengkap maka sistem akan menampilkan halaman registrasi dan menampilkan pesan <i>error</i> seperti "kolom wajib diisi".
<i>Post Condition</i>	Sistem akan menampilkan pesan bahwa registrasi berhasil dilakukan dan menampilkan halaman <i>login</i> .

Tabel 4.6 Skenario *Use Case* Login

<i>SRS-F-200</i>	Login
<i>Objective</i>	Agar pengguna dapat masuk ke dalam sistem
<i>Actor</i>	Pengguna
<i>Pre Condition</i>	Halaman <i>login</i> telah terbuka dan tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. <i>Actor</i> mengisi formulir <i>login</i> dengan <i>username</i> dan <i>password</i> kemudian menekan tombol masuk.

	2. Sistem akan melakukan pengecekan data <i>username</i> dan <i>password</i> yang dimasukkan <i>actor</i> pada <i>database</i> .
Alternative Flow	<ol style="list-style-type: none"> 1. Apabila <i>username</i> dan <i>password</i> belum dimasukkan oleh <i>actor</i> maka sistem akan menampilkan pesan “kolom <i>username</i> wajib diisi, kolom <i>password</i> wajib diisi!”. 2. Apabila <i>username</i> dan <i>password</i> yang dimasukkan salah maka sistem akan menampilkan pesan “Gagal login, cek <i>username</i> dan <i>password</i> anda!” 3. Apabila <i>username</i> yang dimasukkan salah maka sistem akan menampilkan pesan” Gagal Login: Cek <i>username</i> dan <i>password</i> anda!”. 4. Apabila <i>password</i> yang dimasukkan salah maka sistem akan menampilkan pesan” Gagal Login: Cek <i>username</i> dan <i>password</i> anda!”.
Post Condition	<i>Actor</i> telah teridentifikasi sebagai pemohon dan sistem menampilkan halaman biodata.

Tabel 4.7 Skenario *Use Case Login Back Office*

SRS-F-300	<i>Login Back Office</i>
Objective	Agar <i>staff back office</i> , kepala pengendalian, kepala bidang, kepala administrasi dan kepala dinas dapat masuk ke dalam sistem
Actor	<i>staff back office</i> , kepala pengendalian, kepala bidang, kepala administrasi dan kepala dinas
Pre Condition	Halaman <i>login</i> telah terbuka dan tersedia
Main Flow	<ol style="list-style-type: none"> 3. <i>Actor</i> mengisi formulir <i>login</i> dengan <i>username</i>, <i>password</i> dan memilih level kemudian menekan tombol masuk. 4. Sistem akan melakukan pengecekan dari data yang dimasukkan <i>actor</i> pada <i>database</i>.
Alternative Flow	5. Apabila <i>username</i> dan <i>password</i> belum dimasukkan oleh <i>actor</i> maka sistem akan menampilkan pesan “kolom <i>username</i> wajib diisi, kolom <i>password</i> wajib diisi!”.

	<p>6. Apabila <i>username</i> dan <i>password</i> yang dimasukkan salah maka sistem akan menampilkan pesan “Gagal login, cek <i>username</i>, <i>password</i> dan level anda!”</p> <p>7. Apabila <i>username</i> yang dimasukkan salah maka sistem akan menampilkan pesan” Gagal Login: Cek <i>username</i>, <i>password</i> dan level anda!”.</p> <p>8. Apabila <i>password</i> yang dimasukkan salah maka sistem akan menampilkan pesan” Gagal Login: Cek <i>username</i>, <i>password</i> dan level anda!”.</p> <p>9. Apabila level yang dipilih tidak sesuai maka sistem akan menampilkan pesan “Gagal login, cek <i>username</i>, <i>password</i> dan level anda!”</p>
Post Condition	Sistem menampilkan halaman daftar pemohon SIUP.

Tabel 4.8 Skenario Use Case Logout

SRS-F-400	Logout
Objective	Agar pemohon, staff <i>back office</i> , kepala pengendalian, kepala bidang, kepala administrasi dan kepala dinas dapat keluar dari sistem.
Actor	Pemohon, staff <i>back office</i> , kepala pengendalian, kepala bidang, kepala administrasi, kepala dinas
Pre Condition	<i>Actor</i> telah <i>login</i> di sistem
Main Flow	<ol style="list-style-type: none"> 1. <i>Actor</i> menekan tombol <i>logout</i>. 2. Sistem menampilkan halaman beranda.
Alternative Flow	-
Post Condition	<i>Actor</i> berhasil keluar dari sistem.

Tabel 4.9 Skenario Use Case Mengupdate Biodata

SRS-F-500	Mengupdate Biodata
Objective	Pemohon dapat memperbarui informasi biodata.

Actor	Pemohon
Pre Condition	<i>Actor</i> telah berhasil <i>login</i> di sistem dan formulir biodata telah tersedia
Main Flow	<ol style="list-style-type: none"> 1. <i>Actor</i> memasukkan nama, jenis kelamin, tempat lahir, tanggal lahir, alamat, nomor telp, pekerjaan, kewarganegaraan dan mengupload foto pada <i>form</i> biodata 2. <i>Actor</i> menekan tombol <i>update</i>. 3. Sistem menyimpan data yang telah di <i>update</i> ke dalam <i>database</i>.
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat data kosong maka sistem menampilkan pesan “kolom wajib diisi!”. 2. Jika foto belum di <i>upload</i> maka akan menampilkan pesan “Gagal! Gambar belum di <i>upload</i>, <i>upload</i> gambar terlebih dahulu”.
Post Condition	Sistem menampilkan halaman biodata.

Tabel 4.10 Skenario *Use Case* Melihat Notifikasi

SRS-F-600	Melihat notifikasi
Objective	Pemohon dapat mengetahui notifikasi yang masuk
Actor	Pemohon
Pre Condition	<i>Actor</i> telah berhasil <i>login</i> dan terverifikasi sebagai pemohon
Main Flow	<ol style="list-style-type: none"> 1. <i>Actor</i> memilih menu dengan ikon <i>bel</i> untuk melihat notifikasi. 2. Sistem menampilkan pemberitahuan yang masuk.
Alternative Flow	Jika tidak ada pemberitahuan yang masuk maka sistem menampilkan “ tidak ada pemberitahuan”
Post Condition	<i>Actor</i> dapat melihat isi pemberitahuan yang masuk.

Tabel 4.11 Skenario *Use Case* Mengajukan Perizinan SIUP

SRS-F-700	Mengajukan perizinan SIUP
Objective	Agar pemohon dapat mengajukan permohonan perizinan SIUP.
Actor	Pemohon
Pre Condition	<i>Actor</i> telah masuk ke halaman data pemohon
Main Flow	<ol style="list-style-type: none"> 1. <i>Actor</i> memilih menu izin SIUP. 2. Sistem menampilkan halaman perizinan SIUP dan menampilkan <i>form</i> pendaftaran perizinan SIUP. 3. <i>Actor</i> melakukan pengisian formulir pendaftaran perizinan. Formulir pendaftaran terdiri dari identitas pemohon, identitas perusahaan, legalitas perusahaan, kekayaan bersih dan saham, jenis barang atau jasa yang diperjualkan, neraca awal perusahaan dan pemohon mengupload berkas perizinan sesuai dengan persyaratan yang telah dicantumkan dalam formulir pengajuan perizinan seperti Akta pendirian perusahaan, KTP, KK, keterangan domisili tempat usaha, NPWP, bukti kepemilikan usaha, IMB, SIUP lama jika perpanjangan, surat keputusan pengesahan perusahaan, kartu peserta BPJS, LKPM. 4. <i>Actor</i> memilih tombol <i>submit</i> untuk mengajukan perizinan. 5. Sistem akan menampilkan pesan "Permohonan Izin Berhasil di Proses" dan menampilkan halaman status perizinan.
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat data yang belum lengkap atau berkas persyaratan belum lengkap maka sistem akan menampilkan pesan "kolom wajib diisi!" pada setiap kolom yang kosong. 2. Jika Gambar belum di <i>upload</i> maka sistem akan menampilkan pesan "Gagal! Gambar

	belum di <i>upload</i> , <i>upload</i> gambar terlebih dahulu”.
Post Condition	<i>Actor</i> berhasil mengajukan permohonan perizinan SIUP dan sistem menampilkan halaman status perizinan.

Tabel 4.12 Skenario *Use Case* Mengajukan Perizinan TDP

SRS-F-800	Mengajukan perizinan TDP
Objective	Agar pemohon dapat mengajukan permohonan perizinan TDP.
Actor	Pemohon
Pre Condition	<i>Actor</i> telah menekan menu perizinan TDP.
Main Flow	<ol style="list-style-type: none"> 1. Sistem menampilkan halaman perizinan TDP dan menampilkan <i>form</i> pendaftaran perizinan TDP. 2. <i>Actor</i> melakukan pengisian formulir pendaftaran perizinan TDP. Formulir pendaftaran terdiri dari identitas pemilik, identitas perusahaan, legalitas perusahaan dan mengupload berkas perizinan sesuai dengan persyaratan. Berkas persyaratan perizinan terdiri dari Akte pendirian / perubahan perusahaan, surat keputusan pengesahan perusahaan, KTP, KK, NPWP, izin usaha, TDP asli jika perpanjangan, izin IMB, SPPL/ UKL-UPL/ AMDAL, Kartu BPJS. 3. <i>Actor</i> memilih tombol <i>submit</i> untuk mengajukan perizinan. 4. Sistem akan menampilkan pesan “Permohonan Izin Berhasil di Proses” dan menampilkan halaman status perizinan.
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat data yang belum diisi atau berkas persyaratan belum lengkap maka sistem akan menampilkan pesan “kolom wajib diisi!” pada setiap kolom yang kosong. 2. Jika gambar gagal di <i>upload</i> maka sistem akan menampilkan pesan “Gagal! Gambar belum di <i>upload</i>, <i>upload</i> gambar terlebih dahulu”.

Post Condition	<i>Actor</i> berhasil mengajukan permohonan perizinan dan sistem menampilkan halaman status perizinan.
-----------------------	--

Tabel 4.13 Skenario *Use Case* Mengajukan Perizinan TDUP

SRS-F-900	Mengajukan perizinan TDUP
Objective	Agar pemohon dapat mengajukan permohonan perizinan TDUP.
Actor	Pemohon
Pre Condition	<i>Actor</i> telah menekan menu pengajuan perizinan.
Main Flow	<ol style="list-style-type: none"> 1. <i>Actor</i> memilih menu TDUP makanan dan minuman atau TDUP wisata. 2. Sistem menampilkan halaman perizinan TDUP dengan menampilkan <i>form</i> pendaftaran perizinan TDUP sesuai dengan izin yang dipilih. 3. <i>Actor</i> melakukan pengisian formulir pendaftaran perizinan. Formulir pendaftaran terdiri dari nama, alamat, bentuk badan usaha, nama perusahaan, jabatan, alamat kantor, jenis usaha dan melakukan upload berkas perizinan sesuai dengan persyaratan yang tercantum. Berkas perizinan tersebut terdiri dari Akte pendirian perusahaan, NPWP perusahaan atau perorangan, bukti hak atas tanah, IMB, atau IPB, HO, SPPL, izin lingkungan dan surat keabsahan atau kebenaran dokumen, surat kuasa jika ada. 4. <i>Actor</i> memilih tombol <i>submit</i> untuk mengajukan perizinan. 5. Sistem akan menampilkan pesan "Permohonan Izin Berhasil di Proses" dan sistem menampilkan halaman status perizinan.
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat data yang belum diisi atau berkas persyaratan belum lengkap maka sistem akan menampilkan pesan "kolom wajib diisi!" pada setiap kolom yang kosong.

	2. Jika gambar gagal di upload maka sistem akan menampilkan pesan “Gagal! Gambar belum di upload, upload gambar terlebih dahulu.
Post Condition	Actor berhasil mengajukan permohonan perizinan dan sistem menampilkan halaman status perizinan.

Tabel 4.14 Skenario *Use Case* Mengajukan Perizinan Reklame

SRS-F-1000	Mengajukan perizinan reklame
Objective	Agar pemohon dapat mengajukan permohonan perizinan reklame.
Actor	Pemohon
Pre Condition	Actor telah menekan tombol Izin Reklame
Main Flow	<ol style="list-style-type: none"> 1. Actor memilih menu jenis reklame yang akan diajukan. Jenis reklame terdiri dari reklame baru, perpanjangan atau reklame isidental. 2. Sistem menampilkan formulir pendaftaran perizinan reklame. 3. Actor melakukan pengisian data pada formulir pendaftaran perizinan reklame. Formulir pendaftaran berisi seperti nama, alamat/ telp, nama perusahaan, jabatan, alamat perusahaan/telp. Kemudian melengkapi izin permohonan yang terdiri dari bentuk reklame, ukuran dan jumlah reklame, ketinggian reklame, lama pemasangan dan lokasi pemasangan. Kemudian mengupload berkas perizinan sesuai dengan persyaratan seperti KTP/SIM, denah lokasi pemasangan, gambar dan kontruksi reklame, surat pernyataan tidak keberatan dari pemilik tanah, perjanjian sewa, sewa tanah pemda/ propinsi, sertifikat kepemilikan tanah dilegalisir dan IMB. Untuk reklame isidental berkas persyaratan yang harus diupload adalah KTP, gambar obyek atau reklame dan denah lokasi.

	<ol style="list-style-type: none"> Setelah data diisikan semua, <i>actor</i> memilih tombol <i>submit</i> untuk mengajukan perizinan reklame. Sistem akan menampilkan pesan bahwa formulir perizinan berhasil diajukan.
Alternative Flow	<ol style="list-style-type: none"> Jika terdapat data yang belum diisi atau berkas persyaratan belum lengkap maka sistem akan menampilkan pesan “kolom wajib diisi!” pada setiap kolom yang kosong. Jika gambar gagal di upload maka sistem akan menampilkan pesan “Gagal! Gambar belum di upload, upload gambar terlebih dahulu.
Post Condition	<i>Actor</i> berhasil mengajukan permohonan perizinan dan sistem menampilkan halaman status perizinan.

Tabel 4.15 Skenario Use Case Melihat Status Perizinan SIUP, TDP, TDUP, Reklame

SRS-F-1100	Melihat status perizinan SIUP, TDP, TDUP, Reklame
Objective	Pemohon dapat mengetahui sampai dimana permohonan perizinan yang diajukan.
Actor	Pemohon
Pre Condition	<i>Actor</i> telah mengajukan permohonan perizinan dan <i>actor</i> telah masuk ke halaman data pemohon.
Main Flow	<ol style="list-style-type: none"> <i>Actor</i> memilih menu permohonan pengajuan. Sistem menampilkan halaman status perizinan.
Alternative Flow	-
Post Condition	Sistem akan menampilkan status perizinan <i>actor</i> .

Tabel 4.16 Skenario *Use Case* Memeriksa Kelengkapan Berkas SIUP, TDP, TDUP, Reklame

<i>SRS-F-1200</i>	Memeriksa kelengkapan berkas SIUP, TDP, TDUP, Reklame
<i>Objective</i>	Agar staff <i>back office</i> dapat memeriksa kelengkapan dokumen pemohon perizinan SIUP, TDP, TDUP dan Reklame
<i>Actor</i>	Staff <i>back office</i>
<i>Pre Condition</i>	<i>Actor</i> telah <i>login</i> di sistem dan sistem menampilkan halaman daftar pemohon SIUP.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. <i>Actor</i> memilih menu permohonan izin SIUP/TDP/TDUP/reklame pada sidebar. 2. Sistem akan menampilkan daftar pemohon yang mengajukan izin. 3. <i>Actor</i> menekan tombol detail untuk melihat data perizinan dan persyaratan perizinan. 4. Sistem akan menampilkan dokumen tersebut dan menampilkan checklist permohonan. 5. <i>Actor</i> akan mengisi form checklist permohonan dan menekan tombol validasi.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	<i>Actor</i> berhasil memeriksa berkas perizinan yang diajukan pemohon dan menampilkan halaman daftar pemohon.

Tabel 4.17 Skenario *Use Case* Membuat SK Perizinan SIUP, TDP, TDUP, Reklame

<i>SRS-F-1300</i>	Membuat SK perizinan SIUP, TDP, TDUP, Reklame
<i>Objective</i>	Agar staff <i>back office</i> dapat membuat SK perizinan untuk pemohon yang mengajukan izin SIUP/ TDP/ TDUP/ Reklame
<i>Actor</i>	Staff <i>back office</i>
<i>Pre Condition</i>	<i>Actor</i> telah menekan tombol validasi dan formulir SK perizinan telah terbuka.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Sistem menampilkan formulir pembuatan SK sesuai dengan izin yang diajukan.

	<p>2. Sistem akan <i>generate</i> data sesuai dengan pemohon yang mengajukan izin. <i>Actor</i> memeriksa kebenaran data tersebut dan memeriksa kesalahan penulisan. <i>Form</i> SK memiliki isi yang berbeda-beda tergantung dari izin yang diajukan. Untuk perizinan reklame formulir surat keputusan reklame berisi nama pemohon, pekerjaan, alamat pemohon, alamat perusahaan, bentuk reklame, ukuran, jumlah, teks, lokasi yang dimohon dan masa berlaku. Untuk SK TDP berisi nama perusahaan, nama pengurus/ penanggungjawab, alamat perusahaan, NPWP, Nomor telepon, kegiatan usaha dan KBLI. Untuk formulir SK TDUP berisi nama perusahaan, alamat perusahaan, nama usaha, alamat usaha, bidang atau jenis usaha, IMB, HO, SPPL dan izin lingkungan. Untuk SK SIUP berisi nama perusahaan, nama penanggungjawab, alamat perusahaan, nomor telepon, kekayaan bersih, kelembagaan, kegiatan usaha (KBLI), barang atau jasa dagangan.</p> <p>3. <i>Actor</i> menekan tombol <i>submit</i>.</p> <p>4. Sistem akan menampilkan pesan "SK berhasil dibuat".</p>
Alternative Flow	-
Post Condition	<i>Actor</i> berhasil membuat SK perizinan dan menampilkan halaman daftar pemohon.

Tabel 4.18 Skenario *Use Case* Mencetak SK Perizinan SIUP, TDP, TDUP, Reklame

SRS-F-1400	Mencetak SK perizinan SIUP, TDP, TDUP, Reklame
Objective	Staff <i>back office</i> dapat mencetak surat keputusan perizinan
Actor	Staff <i>back office</i>
Pre Condition	Kepala dinas telah membuka halaman cetak SK
Main Flow	1. Sistem menampilkan daftar pemohon.

	2. <i>Actor</i> menekan tombol cetak. 3. Sistem menampilkan dokumen cetak.
Alternative Flow	Jika tidak ada SK yang diterbitkan maka sistem akan menampilkan pesan “ <i>No data available in table</i> ”.
Post Condition	<i>Actor</i> berhasil mencetak surat keputusan perizinan.

Tabel 4.19 Skenario *Use Case* Memberikan Nomor Surat

SRS-F-1500	Memberikan nomor surat
Objective	Agar kepala administrasi dapat memberikan nomor surat pada surat keputusan perizinan.
Actor	Kepala administrasi
Pre Condition	<i>Actor</i> telah berhasil <i>login</i> dan telah masuk halaman daftar pemohon SIUP
Main Flow	1. Sistem menampilkan halaman daftar pemohon sesuai dengan kategori perizinan. 2. <i>Actor</i> memilih tombol detail untuk mengedit SK perizinan. 3. Sistem menampilkan <i>form</i> SK perizinan. 4. <i>Actor</i> mengedit SK perizinan pada bagian nomor SK dan memberikan nomor untuk SK perizinan. 5. <i>Actor</i> menekan tombol <i>update</i> . 6. Sistem menampilkan halaman daftar pemohon izin.
Alternative Flow	-
Post Condition	<i>Actor</i> berhasil memberikan nomor pada SK perizinan.

Tabel 4.20 Skenario *Use Case* Menvalidasi Berkas Perizinan

SRS-F-1600	Menvalidasi berkas perizinan SIUP, TDP, TDUP, Reklame
Objective	Agar kepala pengendalian dan kepala bidang dapat menvalidasi berkas perizinan sesuai dengan persyaratan.
Actor	Kepala pengendalian, kepala bidang

Pre Condition	<i>Actor</i> telah <i>login</i> di sistem dan <i>actor</i> telah masuk ke halaman daftar pemohon.
Main Flow	<ol style="list-style-type: none"> 1. <i>Actor</i> memilih menu daftar pemohon perizinan sesuai jenis izin yang ingin di validasi 2. Sistem akan menampilkan daftar pemohon yang mengajukan izin. 3. <i>Actor</i> tombol details. 4. Sistem akan menampilkan dokumen pemohon beserta berkas yang diajukan. 5. <i>Actor</i> memeriksa kelengkapan data dan berkas yang di <i>upload</i> pemohon dengan memberikan tanda checklist dan menekan tombol validasi. 6. Sistem menampilkan pesan pemberitahuan bahwa data berhasil di validasi.
Alternative Flow	-
Post Condition	<i>Actor</i> berhasil memvalidasi berkas pemohon.

Tabel 4.21 Skenario *Use Case* Menerbitkan SK perizinan SIUP, TDP, TDUP, Reklame

SRS-F-1700	Menerbitkan SK perizinan SIUP, TDP, TDUP, Reklame
Objective	Agar kepala dinas dapat menerbitkan SK perizinan bagi pemohon.
Actor	Kepala dinas
Pre Condition	<i>Actor</i> telah <i>login</i> di sistem dan SK perizinan telah dibuat oleh staff <i>back office</i> . <i>Actor</i> telah masuk ke halaman daftar pemohon
Main Flow	<ol style="list-style-type: none"> 1. <i>Actor</i> memilih menu daftar pemohon perizinan sesuai izin yang diajukan. 2. Sistem akan menampilkan daftar pemohon. 3. <i>Actor</i> menekan tombol terbitkan untuk menerbitkan SK perizinan. 4. Sistem akan menampilkan pesan "Data pemohon berhasil di terbitkan".
Alternative Flow	-
Post Condition	<i>Actor</i> berhasil menerbitkan SK perizinan.

Tabel 4.22 Skenario *Use Case* Melihat Daftar Pemohon

SRS-F-1800	Melihat daftar pemohon
Objective	Agar kepala pengendalian, kepala bidang dan kepala dinas dapat melihat pemohon yang terdaftar di sistem.
Actor	Kepala pengendalian, kepala bidang, kepala dinas.
Pre Condition	<i>Actor</i> telah <i>login</i> di sistem dan telah menekan tombol daftar pemohon.
Main Flow	<ol style="list-style-type: none"> 1. Sistem menampilkan halaman daftar pemohon. 2. <i>Actor</i> menekan tombol details 3. Sistem akan menampilkan identitas pemohon dan izin apa yang diajukan.
Alternative Flow	Apabila tidak ada pemohon yang terdaftar di sistem maka sistem menampilkan pesan bahwa “No data available in table”.
Post Condition	<i>Actor</i> berhasil melihat pemohon yang terdaftar di sistem.

Tabel 4.23 Skenario *Use Case* Menghapus Akun Pemohon

SRS-F-1900	Menghapus akun pemohon
Objective	Kepala pengendalian dapat menghapus akun pemohon yang terdaftar di sistem.
Actor	Kepala pengendalian
Pre Condition	<i>Actor</i> telah <i>login</i> di sistem dan telah membuka halaman daftar pemohon.
Main Flow	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar pemohon. 2. <i>Actor</i> memilih tombol hapus berupa ikon silang. 3. Sistem akan menampilkan pesan pop-up terlebih dahulu dengan pertanyaan “Apakah anda yakin akan menghapus akun ini?” 4. <i>Actor</i> menekan tombol OK. 5. Sistem menampilkan pesan “Data pemohon berhasil dihapus”.

Alternative Flow	<ol style="list-style-type: none">1. Actor dapat membatalkan pemohon yang akan dihapus dengan menekan tombol <i>cancel</i>.2. Apabila data gagal dihapus maka akan menampilkan pesan "Data gagal di hapus".
Post Condition	Actor berhasil menghapus akun pemohon.











BAB 5 PERANCANGAN DAN IMPLEMENTASI

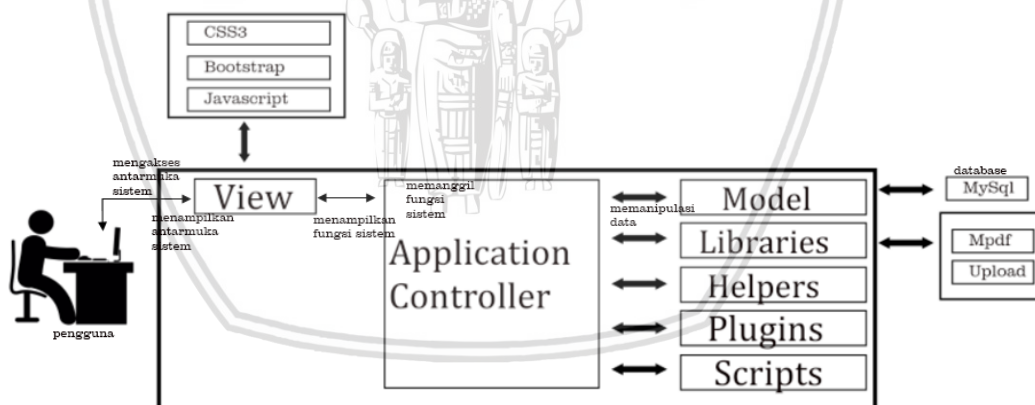
Perancangan dibuat untuk menggambarkan model dari sistem yang dibuat. Dalam Bab ini akan menjabarkan perancangan dan implementasi dari Sistem Perizinan.

5.1 Perancangan Sistem

Perancangan merupakan tahapan yang dilakukan setelah rekayasa kebutuhan. Tahapan ini menggambarkan rancangan sistem dan interaksi objek dengan sistem. Pada penelitian ini, perancangan yang dilakukan yaitu perancangan arsitektur, pemodelan diagram *sequence*, pemodelan diagram kelas, perancangan data dan perancangan antarmuka.

5.1.1 Perancangan Arsitektur Sistem

Pada Gambar 5.1 digambarkan perancangan arsitektur dari sistem perizinan yang menggunakan pendekatan *Model View Controller* (MVC). Pengguna akan mengakses *view* melalui browser yang akan menampilkan halaman antarmuka sistem. Halaman *view* sendiri akan mengakses *application controller* yang terdiri dari *model*, *libraries*, *helpers*, *plugins* dan *script* yang dibutuhkan sesuai permintaan. Dalam *application controller* akan memanipulasi data menggunakan model. Model itu sendiri terhubung dengan MySQL *database* untuk penyimpanan data.

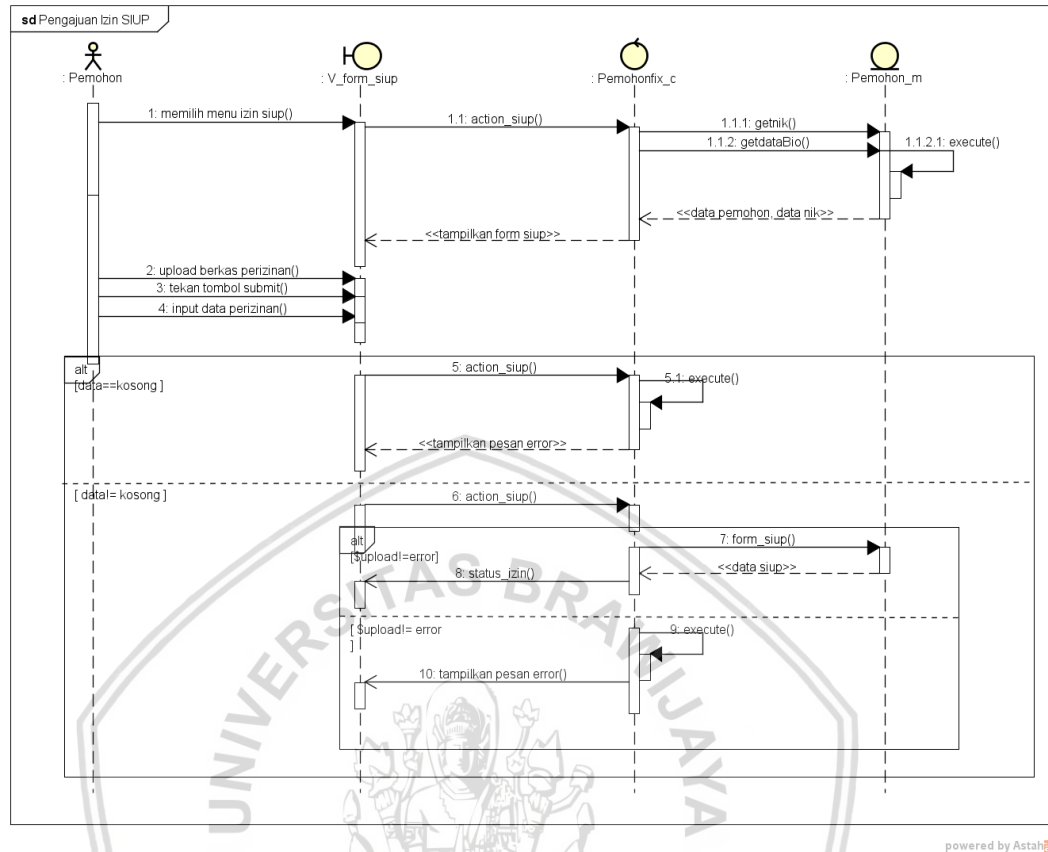


Gambar 5.1 Perancangan Arsitektur Sistem Perizinan

5.1.2 Pemodelan Diagram *Sequence*

Diagram *sequence* menggambarkan interaksi antar objek yang ada di sistem. Diagram *sequence* yang digambarkan dalam penelitian ini ada empat diagram yang meliputi proses pengajuan izin SIUP, proses validasi perizinan, proses *update* biodata dan proses menghapus pemohon.

5.1.2.1 Diagram Sequence Pengajuan Izin SIUP



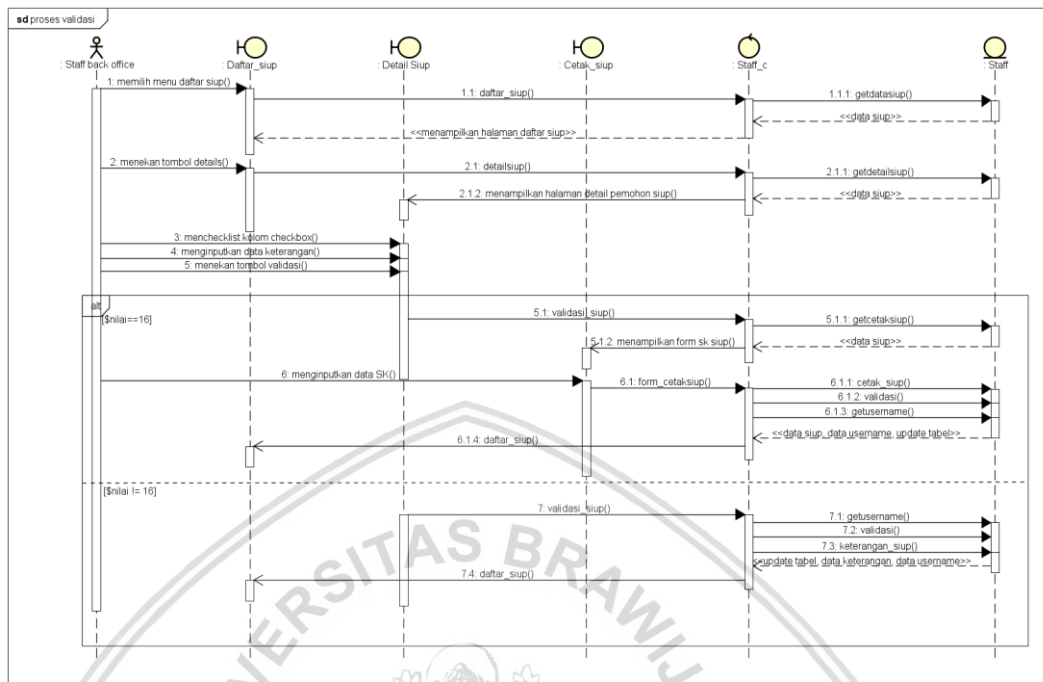
Gambar 5.2 Pengajuan Izin SIUP

Pada Gambar 5.2 menunjukkan alur dari fungsi pengajuan izin SIUP yang dimulai dari pemohon memilih menu izin SIUP direspon oleh sistem dengan menampilkan halaman formulir izin SIUP. Kemudian pemohon mengisi formulir izin SIUP, mengupload berkas dokumen dan menekan tombol *submit*. Setelah berhasil di *upload* maka sistem akan menyimpan data ke dalam *database* dan sistem menampilkan halaman status izin. Apabila terdapat kolom kosong yang belum diisi pemohon maka sistem akan menampilkan pesan kesalahan. Pada diagram *use case* proses ini diberikan nama mengajukan perizinan siup.

5.1.2.2 Diagram Sequence Proses Validasi Perizinan

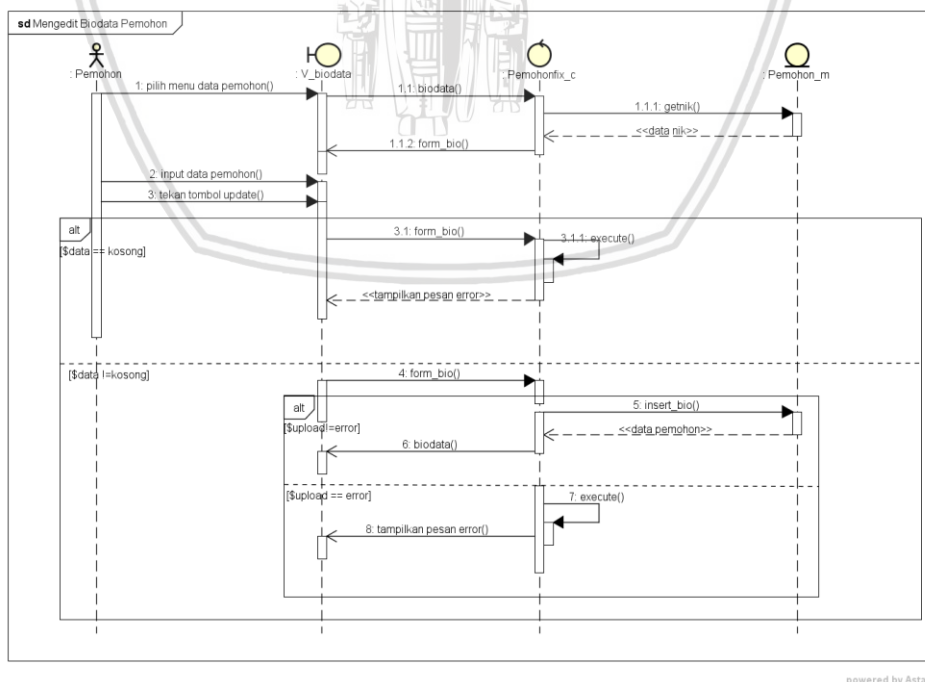
Pada Gambar 5.3 menggambarkan alur dari proses validasi perizinan. Proses ini di dalam *use case* diberikan nama memeriksa kelengkapan berkas perizinan SIUP, TDP, TDUP, Reklame. Proses dimulai dari Staffback memilih menu daftar SIUP dan sistem menampilkan halaman daftar pemohon SIUP. Kemudian staff *back office* memilih tombol detail dan sistem menampilkan halaman detail pemohon. staff *back office* memvalidasi berkas pemohon dengan menchecklist pada kolom *checkbox*, mengisi form keterangan dan menekan tombol validasi. Apabila data yang dimasukkan pemohon valid maka sistem akan menampilkan formulir SK pemohon dan staff *back office* akan menginputkan data pemohon.

Kemudian sistem menyimpan data tersebut ke *database*. Jika data pemohon tidak valid maka sistem akan menampilkan halaman daftar pemohon SIUP.



Gambar 5.3 Menvalidasi Berkas SIUP

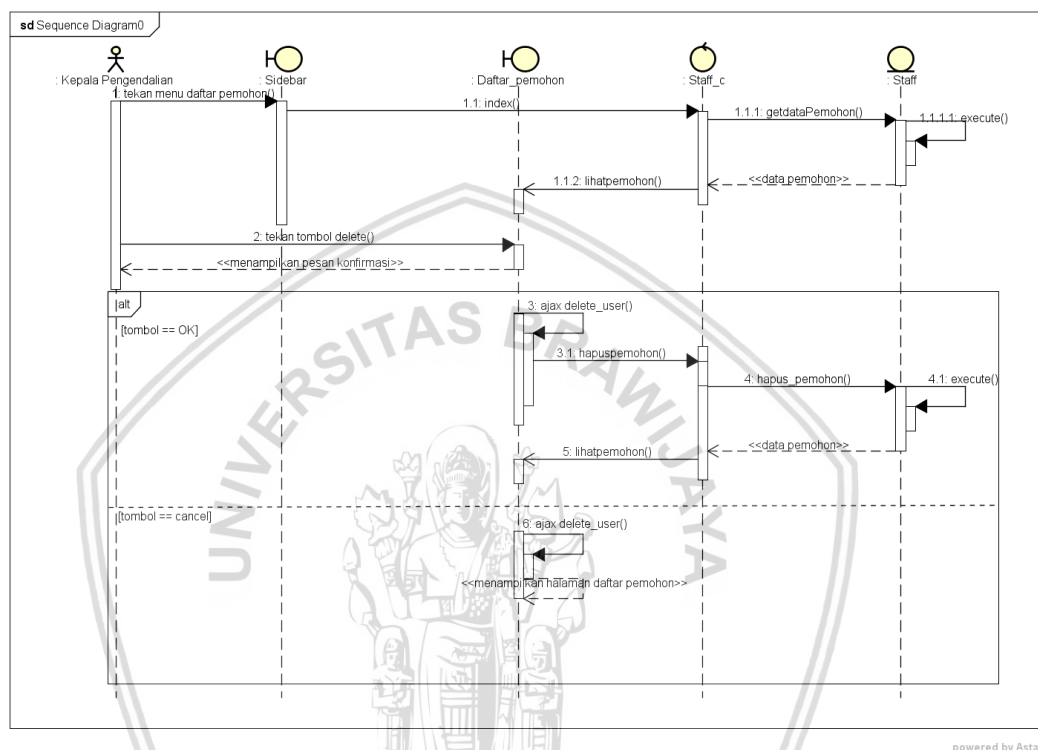
5.1.2.3 Diagram Sequence Update Biodata



Gambar 5.4 Update Biodata

Pada Gambar 5.4 merepresentasikan alur dari fungsi update biodata. Proses dimulai dari pemohon memilih menu daftar pemohon dan sistem menampilkan halaman formulir biodata yang sebelumnya telah terisi. Pemohon mengedit data yang ingin di *update* dan menekan tombol *update*. Selanjutnya sistem akan mengupdate *database* dan menampilkan halaman biodata. Pada diagram *use case* proses ini diberikan nama mengupdate biodata.

5.1.2.4 Diagram Sequence Menghapus Pemohon



Gambar 5.5 Menghapus Pemohon

Pada Gambar 5.5 merepresentasikan alur dari fungsi menghapus pemohon. Proses dimulai ketika kepala pengendalian memilih menu daftar pemohon dan sistem menampilkan daftar pemohon. Kemudian kepala pengendalian menekan tombol hapus. Sistem menampilkan pesan konfirmasi. Jika kepala pengendalian memilih "OK" maka sistem akan menghapus data pemohon yang ada di *database* dan menampilkan pesan data berhasil dihapus. Jika kepala pengendalian memilih "cancel" maka disistem menampilkan halaman daftar pemohon. Pada diagram *use case* proses ini diberikan nama menghapus akun pemohon.

5.1.3 Pemodelan Diagram Klas

Diagram klas merupakan diagram yang menggambarkan struktur dan deskripsi klas beserta dengan relasi-relasi yang ada. Diagram klas berfungsi untuk menjelaskan tipe dari objek sistem dan hubungannya dengan objek lain. Diagram klas dalam penelitian ini ditunjukkan pada Gambar 5.6.

Pemodelan diagram klas yang ditunjukkan pada Gambar 5.6 merupakan pemodelan diagram klas dengan struktur MVC atau *Model*, *View* dan *Controller*. Klas model dalam sistem ini terdiri dari 2 klas yaitu klas *staff_m* dan klas *pemohon* yang sama-sama memiliki hubungan generalisasi dengan klas *CI_Model*. Klas *View* tidak memiliki hubungan generalisasi dengan klas manapun. Untuk klas *controller* sendiri terdiri dari klas *register_c*, *login_c*, *pemohonfix_c* dan klas *staff_c*. keempat klas *controller* tadi memiliki hubungan generalisasi dengan klas *CI_Controller*.

Setelah melakukan pemodelan diagram klas, lalu akan dijelaskan algoritme pada operasi yang ada pada diagram klas. Algoritme pada operasi akan menunjukkan bagaimana alur yang dijalankan sistem untuk dapat menghasilkan suatu keluaran. Perancangan algoritme operasi dibuat hanya beberapa sampel operasi saja yaitu operasi *action_siup* pada klas *Pemohonfix_c*, operasi, *daftar_siup* pada klas *Staff_c*, *biodata* pada klas *Pemohonfix_c* dan *lihatpemohon* pada klas *Staff_c*.

1. Nama klas : *Pemohonfix_c*
 Nama Operasi : *action_siup*
 Algoritme

Tabel 5.1 Algoritme Action_siup

```

If (formvalidation == false) then
    menampilkan pesan error
    menampilkan halaman form siup
else
    memanggil tabel siup
    memasukkan identitas perusahaan, akta pendirian, akta perubahan ke
    dalam array dengan nama data
    inialisasi variabel upload_1, upload_2, upload_3, upload_4, upload_5,
    upload_6, upload_7, upload_8, upload_9, upload_10, upload_11,
    upload_12, upload_13, upload_14, upload_15, upload_16
    memasukkan data gambar ke folder upload
    If(upload!=error) then
        memanggil model form_siup dari klas pemohon
        menyimpan array data dan data gambar ke database
        menampilkan pesan sukses
        menampilkan halaman status_izin
    else
        menampilkan pesan gagal
  
```

```

menampilkan halaman form siup
End if
End if

```

2. Nama Klas : Staff_c
 Nama Operasi : daftar_siup
 Algoritme

Tabel 5.2 Algoritme Daftar_siup

```

inisialisasi variabel username = session [logged_in]
inisialisasi variabel data = mengambil semua data dengan memanggil
operasi getdatasiup()
if (username == 'staffback')
    id_status = "1"
    menampilkan halaman daftar siup
else if (username == 'pengendalian')
    id_status = "2"
    menampilkan halaman daftar siup
else if (username == 'kepalabidang')
    id_status = "3"
    menampilkan halaman daftar siup
else if ()
    id_status = "5"
    menampilkan halaman daftar siup
else
    id_status = "4"
    menampilkan halaman daftar siup
endif

```

3. Nama klas :Pemohonfix_c
 Nama Operasi : biodata
 Algoritme

Tabel 5.3 Algoritme Biodata

```

inisialisasi array data[query] = ambil semua data dengan memanggil operasi
getdataBio()

inisialisasi array data [pemberitahuan] = ambil semua data dengan
memanggil operasi getPemberitahuan()

menampilkan halaman tampil_bio

```

4. Nama Klas : Staff_c
 Nama Operasi : lihatpemohon
 Algoritme

Tabel 5.4 Algoritme Lihatpemohon

```

mengambil nilai username dari session login
mendapatkan nilai username dari model getUsername
inisialisasi variabel data = ambil semua data pada operasi
getdatapemohon()

if (user == 'pengendalian') then
    menampilkan halaman hapus pemohon
else if (user == 'kepalabidang') then
    menampilkan halaman daftar pemohon
else
    menampilkan halaman akses
end if

```

5.1.4 Perancangan Basis Data

Perancangan basis data yang digunakan dalam sistem ini yaitu menggunakan diagram ERD yang memperlihatkan relasi atau hubungan antar entitas yang ada di dalam sistem perizinan. Perancangan ERD ditunjukkan pada Gambar 5.7. Setelah membuat ERD, maka selanjutnya dibuat *Conceptual Data Model* (CDM) yang merupakan penjabaran dari ERD dan terakhir membuat *Physical Data Model* (PDM) untuk merancang basis data secara fisik yang nantinya dijadikan acuan dalam pembuatan basis data sistem

Pada model ERD yang terdapat pada Gambar 5.7 terdapat 10 entitas kuat, 16 entitas lemah dan 2 atribut *multivalued*. Entitas kuat merupakan entitas yang dapat berdiri sendiri tanpa bergantung dengan entitas lain sedangkan entitas kuat selalu bergantung dengan entitas lainnya. Yang termasuk ke dalam entitas kuat

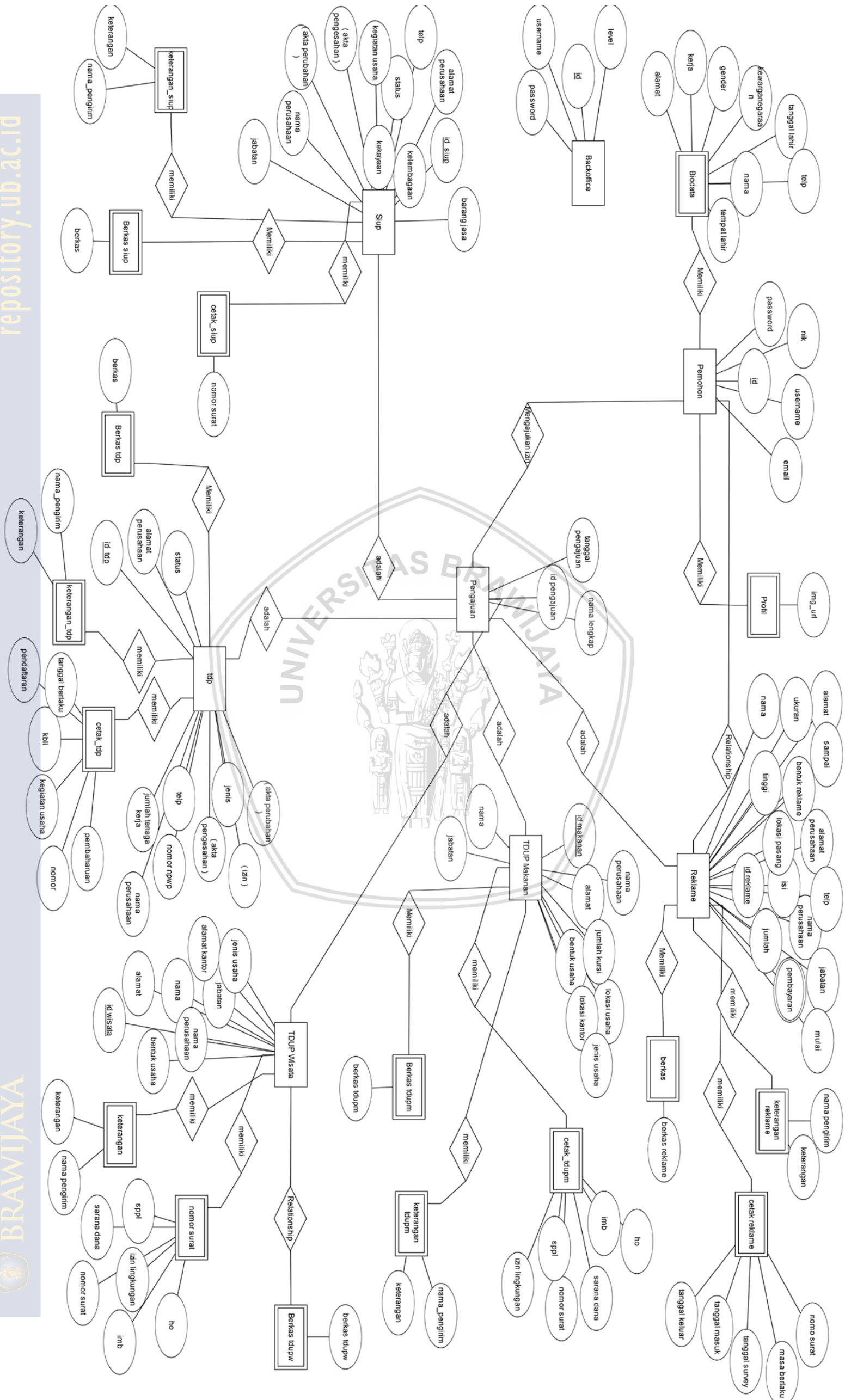
adalah *back office*, pemohon, pengajuan, siup, tdp, tdup makanan, tdup wisata, reklame, jenis_perizinan dan status_pemohon. Sedangkan untuk entitas lemah yaitu:

1. Biodata_pemohon bergantung pada entitas pemohon.
2. Berkas_siup, cetak_siup dan keterangan_siup bergantung pada entitas Siup.
3. Berkas_tdp, cetak_tdp dan keterangan_tdp bergantung pada entitas TDP.
4. Berkas_tdupm, cetak_tdupm dan keterangan_tdupm bergantung pada entitas TDUP Makanan.
5. Berkas_tdupw, cetak_tdupw dan keterangan_tdupw bergantung pada entitas TDUP Wisata.
6. Berkas_reklame, cetak_reklame dan keterangan_reklame bergantung pada entitas Reklame.

Untuk atribut *multivalued* adalah pembayaran yang berasal dari entitas reklame, atribut fotoprofil yang berasal dari entitas pemohon.

Dalam diagram ERD untuk mengubah menjadi CDM maka Entitas yang kuat dan lemah dibuat tabel sendiri beserta atributnya. Atribut *multivalued* dijadikan tabel tersendiri juga dengan nama atribut yaitu id dari entitas utamanya dan nama dari atribut *multivalued* itu sendiri. Hasil implementasi CDM dapat dilihat pada Gambar 5.8 yang menghasilkan 28 tabel.

Selanjutnya yaitu implementasi dari CDM menjadi PDM untuk merancang secara fisik basis data yang nantinya akan dijadikan acuan dalam implementasi basis data sistem. Hasil dari CDM tadi disesuaikan nama tabel beserta atribut yang mengikutinya dan diberikan tipe data untuk setiap atribut. Relasi yang terdapat dalam CDM juga harus diikutsertakan dalam implementasi PDM. Hasil dari implementasi PDM dapat dilihat pada Gambar 5.9.



Gambar 5.7 Entity Relationship Diagram

5.1.5.1 Antarmuka Pengajuan Izin SIUP

Kekayaan Bersih	<input type="text"/>
Kelembagaan	<input type="text"/>
Kegiatan Usaha	<input type="text"/>
Barang dan Jasa	<input type="text"/>
Akta Pendirian/perubahan perusahaan	<input type="text"/> Browse... No file selected.
SKPP	<input type="text"/> Browse... No file selected.
Akta Pendirian/perubahan perusahaan di PN	<input type="text"/> Browse... No file selected.
KK / KTP	<input type="text"/> Browse... No file selected.
Keterangan Domisili Tempat Usaha	<input type="text"/> Browse... No file selected.
NPWP / NPWD	<input type="text"/> Browse... No file selected.
Sertifikat Tanah	<input type="text"/> Browse... No file selected.
Neraca Awal Perusahaan	<input type="text"/> Browse... No file selected.
IMB	<input type="text"/> Browse... No file selected.
SPPL/ UKL/ Dokumen Ijin Lingkungan lainnya	<input type="text"/> Browse... No file selected.
SIUP Lama Asli (untuk perpanjangan)	<input type="text"/> Browse... No file selected.
Pas Photo Background Berwarna	<input type="text"/> Browse... No file selected.
Laporan Kegiatan Penanaman Modal	<input type="text"/> Browse... No file selected.
Kartu Kepesertaan BPJS	<input type="text"/> Browse... No file selected.

Perancangan antarmuka dari halaman form SIUP direpresentasikan pada Gambar 5.10. Halaman ini berisi formulir yang digunakan untuk mengajukan izin oleh pemohon. Formulir tersebut terdiri dari kolom untuk identitas perusahaan pemohon dan kolom untuk *upload* dokumen persyaratan perizinan. Ada 16 kolom *upload* yang masing-masing disediakan untuk persyaratan dokumen. Tiap kolom *upload* berisi file yang berbeda-beda sesuai jenis persyaratannya.

5.1.5.2 Antarmuka Proses Validasi Perizinan

Perancangan antarmuka validasi perizinan direpresentasikan pada Gambar 5.11. Halaman ini menampilkan identitas pemohon, identitas perusahaan, legalitas perusahaan dan dokumen persyaratan yang telah di upload oleh pemohon.

Halaman Staffback

Daftar Pemohon SIUP

Daftar Pemohon TDP

Daftar Pemohon TDUP ✓

Daftar Pemohon Reklame

Daftar Pemohon

Cetak SK

logout

DETAIL PEMOHON

A. Identitas Pemohon

B. Identitas Perusahaan

C. Legalitas Perusahaan

D. Berkas yang dilampirkan

Persyaratan	Dokumen	Sesuai
Akta Pendirian/perubahan perusahaan	<input type="checkbox"/>	<input type="checkbox"/>
SKPP	<input type="checkbox"/>	<input type="checkbox"/>
Akta Pendirian/perubahan perusahaan di PN	<input type="checkbox"/>	<input type="checkbox"/>
KK / KTP	<input type="checkbox"/>	<input type="checkbox"/>
Keterangan Domisili Tempat Usaha	<input type="checkbox"/>	<input type="checkbox"/>
NPWP / NPWD	<input type="checkbox"/>	<input type="checkbox"/>
Sertifikat Tanah	<input type="checkbox"/>	<input type="checkbox"/>
Neraca Awal Perusahaan	<input type="checkbox"/>	<input type="checkbox"/>
IMB	<input type="checkbox"/>	<input type="checkbox"/>
SPPL / UKL / Dokumen Ijin Lingkungan lainnya	<input type="checkbox"/>	<input type="checkbox"/>
SIUP Lama Asli (untuk perpanjangan)	<input type="checkbox"/>	<input type="checkbox"/>
Pas Photo Background Berwarna	<input type="checkbox"/>	<input type="checkbox"/>
Laporan Kegiatan Penanaman Modal	<input type="checkbox"/>	<input type="checkbox"/>
Kartu Kepesertaan BPJS	<input type="checkbox"/>	<input type="checkbox"/>

Keterangan

Validasi

Gambar 5.11 Perancangan Antarmuka Halaman Validasi

5.1.5.3 Antarmuka Update Biodata

Pada Gambar 5.12 merepresentasikan perancangan antarmuka update biodata. Halaman ini terdiri dari kolom identitas pemohon yaitu kolom NIK, nama pemohon, jenis kelamin yang direpresentasikan dengan *radio button*, kolom tempat lahir, tanggal lahir, alamat, nomor telp, pekerjaan dan kewarganegaraan direpresentasikan dengan *choose selection* dan *upload* fotoprofil yang direpresentasikan dengan ikon *browse*.

Gambar 5.12 Perancangan Antarmuka Halaman Update Biodata

5.1.5.4 Antarmuka Menghapus Pemohon

Pada Gambar 5.13 merepresentasikan perancangan antarmuka menghapus pemohon. Halaman ini menampilkan data pemohon yang terdaftar di sistem dalam bentuk tabel. Tabel tersebut berisi *username*, NIK, izin yang diajukan dan tombol hapus yang direpresentasikan dengan ikon silang (x).

Halaman Pengendalian

Daftar Pemohon SIUP
Daftar Pemohon TDP
Daftar Pemohon TDUP
Daftar Pemohon Reklame
Daftar Pemohon
Cetak SK

logout

DAFTAR PEMOHON

	Nama Pemohon	Username	NIK	Details	Delete
1	Hindun Iva	Hinduniva	5760012001	details	X
2	AnisahDenis	anisahde	5760012002	details	X
3	Rery M	rery	5760012003	details	X
4	Hamzah	Hamzah	5760012004	details	X

Gambar 5.13 Perancangan Antarmuka Halaman Menghapus Pemohon

5.2 Implementasi Sistem

Implementasi merupakan tahapan yang dilakukan setelah perancangan. Perancangan yang telah dibuat akan diimplementasikan ke dalam kode program. Pembahasan implementasi terdiri dari spesifikasi sistem, batasan implementasi, implementasi kode program dan implementasi antarmuka.

5.2.1 Spesifikasi Sistem

Hasil dari perancangan yang telah diuraikan pada Bab sebelumnya menjadi acuan dalam melakukan implementasi sistem. Spesifikasi sistem diimplementasikan pada spesifikasi perangkat keras dan perangkat lunak.

5.2.1.1 Spesifikasi Perangkat Keras

Dalam pembangunan Sistem Perizinan, pembuatan sistem ini menggunakan sebuah komputer dengan spesifikasi sebagai berikut yang ditunjukkan pada Tabel 5.5

Tabel 5.5 Spesifikasi Perangkat Keras Komputer

No	Nama Komponen	Spesifikasi
1.	<i>Model Sistem</i>	<i>Asus series A455L</i>
2.	<i>Processor</i>	<i>Intel® Core™ i5-5200U CPU @2.20GHZ (4 CPUs), ~2.2GHz</i>
3.	<i>Memori (RAM)</i>	<i>8 GB</i>
4.	<i>Harddisk</i>	<i>1 TB</i>
5.	<i>Kartu Grafik</i>	<i>Intel®HD Graphics 5500</i>

5.2.1.2 Spesifikasi Perangkat Lunak

Pada pembuatan sistem perizinan menggunakan spesifikasi perangkat lunak yang ditunjukkan pada Tabel 5.6

Tabel 5.6 Spesifikasi Perangkat Lunak

No	Nama Komponen	Spesifikasi
1.	Sistem Operasi	<i>Windows 10 Pro 64-bit (10.0, Build 15063)</i>
2.	Bahasa Pemrograman	<i>Java, HTML, PHP, CSS</i>
3.	Editor	<i>Sublime Text 3, Notepad++</i>
4.	Browser	<i>Google Chrome, Firefox</i>

5.2.2 Batasan Implementasi

Batasan implementasi yang digunakan dalam sistem perizinan yaitu:

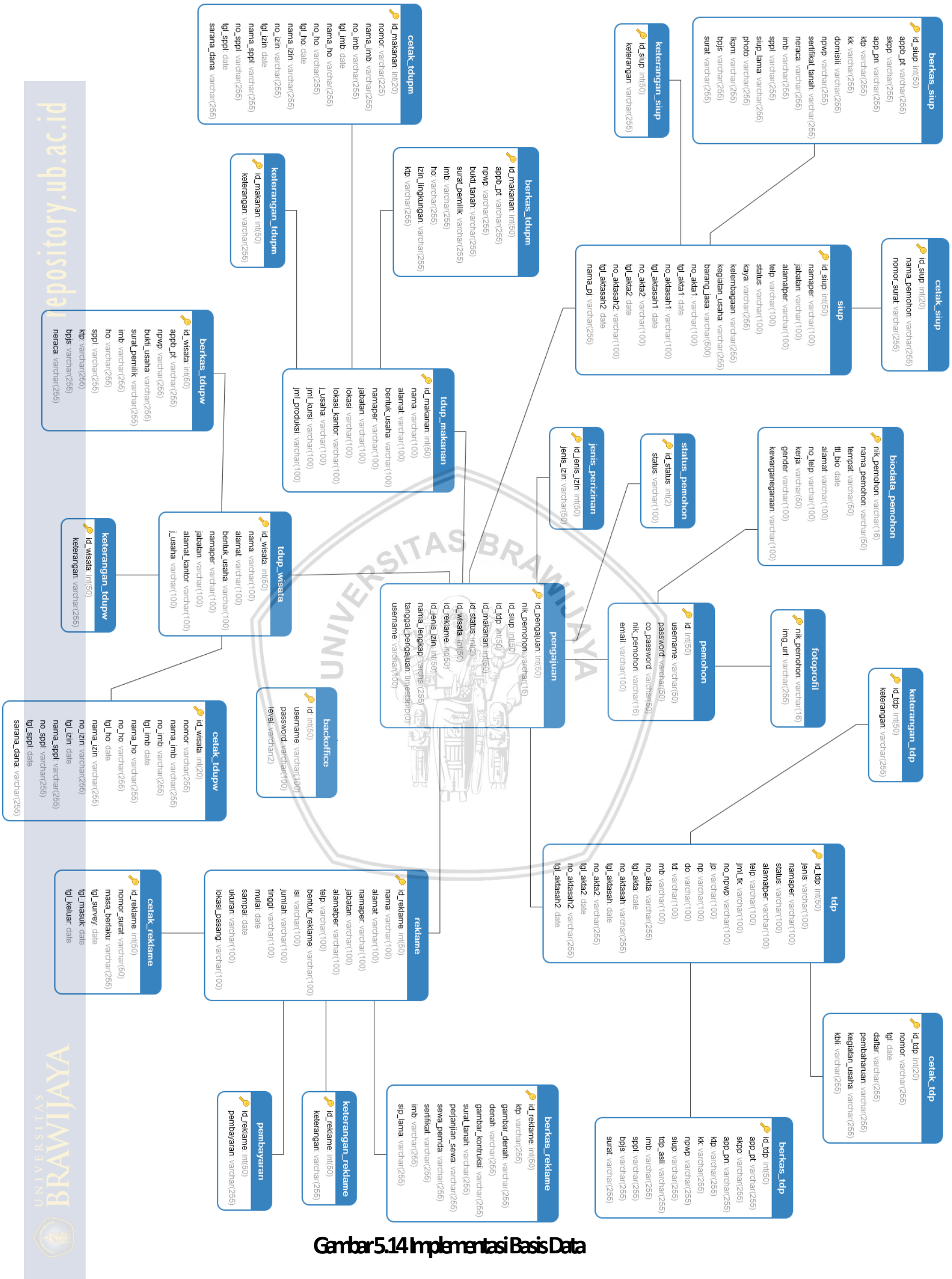
1. Pembangunan sistem menggunakan bahasa pemrograman PHP dengan *framework codeigniter*.

2. Ada dua halaman web yang terdiri dari halaman pemohon dan halaman *back office*.
3. Penyimpanan data untuk sistem perizinan menggunakan *database* MySQL.
4. Implementasi antarmuka menggunakan *html*, *bootstrap*, *css* dan *javascript*.

5.2.3 Implementasi Basis Data

Pada Gambar 5.14 merupakan gambaran basis data dari sistem perizinan yang diimplementasikan menggunakan database MySQL. Implementasi basis data dibuat berdasarkan dari hasil perancangan *physical data model* yang telah dibuat pada bagian perancangan. Dalam sistem ini terdapat 28 tabel yang terdiri dari tabel *back office*, pemohon, biodata_pemohon, profil, pengajuan, status_pemohon, jenis_perizinan, siup, berkas_siup, cetak_siup, keterangan_siup, tdp, berkas_tdp, cetak_tdp, keterangan_tdp, tdup_makanan, berkas_tdupm, cetak_tdupm, keterangan_tdupm, tdup_wisata, berkas_tdupw, cetak_tdupw, keterangan_tdupw, reklame, berkas_reklame, cetak_reklame, keterangan_reklame dan pembayaran.





Gambar 5.14 Implementasi Basis Data

5.2.4 Implementasi Kode Program

5.2.4.1 Implementasi Kode Program Pengajuan Perizinan SIUP

Pada Tabel 5.7 merupakan implementasi dari kode program Pengajuan Perizinan SIUP yang ada pada method action_siup di dalam klas *controller* pemohonfix_c.

Tabel 5.7 Kode Program Action_siup

```
if ($this->form_validation->run() == FALSE){
    $user = $this->session->userdata('username');
    $data['pengguna']=$this->pemohon->getnik($user);
    $this->load->view('pemohon/sidebar_pemohon',$data);
    $this->load->view('pemohon/siup/formsiup',$data);}else{
    $data = array(
        'namaper'=>$this->input->post('namaper'),
        'jabatan'=>$this->input->post('jabatan'),
        'alamatper'=>$this->input->post('alamatper'),
        'telp'=>$this->input->post('telp'),
        'status'=>$this->input->post('statusper'),
        'kaya'=>$this->input->post('kaya'),
        'kelembagaan'=>$this->input->post ('kelembagaan'),
        'kegiatan_usaha'=> $this->input->post('kegiatan_usaha'),
        'barang_jasa'=> $this->input->post('barang_jasa'),
        'no_akta1'=> $this->input->post('no_ak'),
        'tgl_akta1'=> $this->input->post('tgl_ak')
        'no_aktasah'=> $this->input->post('no_sah'),
        'tgl_aktasah'=> $this->input->post('tgl_sah'),
        'no_akta2'=> $this->input->post('no_ak2'),
        'tgl_akta2'=>$this->input->post('tgl_ak2'),
        'no_aktasah2'=>$this->input->post('no_sah2'),
        'tgl_aktasah2'=>$this->input->post('tgl_sah2'),
        'nama_pj'=>$this->input->post('namap'));
    $upload_1      =$this->image_upload->upload('gambar1',$this->directory3);
    $upload_2      =$this->image_upload->upload('gambar2',$this->directory3);
    $upload_3      =$this->image_upload->upload('gambar3',$this->directory3);
    $upload_4      =$this->image_upload->upload('gambar4',$this->directory3);
```

```
$upload_5      =$this->image_upload->upload('gambar5',$this->directory3);
$upload_6      =$this->image_upload->upload('gambar6',$this->directory3);
$upload_7      =$this->image_upload->upload('gambar7',$this->directory3);
$upload_8      =$this->image_upload->upload('gambar8',$this->directory3);
$upload_9      =$this->image_upload->upload('gambar9',$this->directory3);
$upload_10     =$this->image_upload->upload('gambar10',$this->directory3);
$upload_11     =$this->image_upload->upload('gambar11',$this->directory3);
$upload_12     =$this->image_upload->upload('gambar12',$this->directory3);
$upload_13     =$this->image_upload->upload('gambar13',$this->directory3);
$upload_14     =$this->image_upload->upload('gambar14',$this->directory3);
$upload_15     =$this->image_upload->upload('gambar15',$this->directory3);
$upload_16     =$this->image_upload->upload('gambar16',$this->directory3);

$result=$this->pemohon->form_siup($data, $params);
    echo '<script>
    alert("Permohonan Izin Berhasil di Proses");
    window.location.href="status_izin";
    </script>';
    }

    else {
        $data['message'] = "Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!";
        $user = $this->session->userdata('username');
        $data['pengguna']=$this->pemohon->getnama ($user);
        $this->load->view('pemohon/sidebar_pemohon', $data);
        $this->load->view('pemohon/siup/formsiup', $data);
    }
}
```

5.2.4.2 Implementasi Kode Program Validasi Perizinan

Pada Tabel 5.8 merupakan implementasi dari kode program proses validasi perizinan yang ada pada klas *controller* *staff_c* dengan mengambil salah satu method yaitu *daftar_siup*

Tabel 5.8 Kode Program Daftar_siup

```
$username = $this->session->userdata('username');
    $data ['user']= $this->staff-
>getusername($username);

    if($username == 'staffback'){
        $id_status = "1";
        $data ['query']=$this->staff->getdatasiup($id_status);
        $this->load->view('backoffice/sidebar',$data);
        $this->load->view('staffpengendalian/daftar_siup',
$data);
    }else if($username == 'pengendalian'){
        $id_status = "2";
        $data ['query']=$this->staff->getdatasiup($id_status);
        $this->load->view('backoffice/sidebar',$data);
        $this->load->view('staffpengendalian/daftar_siup',
$data);
    }else if($username == 'kepalabidang'){
        $id_status = "3";
        $data ['query']=$this->staff->getdatasiup($id_status);
        $this->load->view('backoffice/sidebar',$data);
        $this->load->view('staffpengendalian/daftar_siup',
$data);
    }else if($username == 'kepalaadmin'){
        $id_status = "5";
        $data ['query']=$this->staff->getdatasiup($id_status);
        $this->load->view('backoffice/sidebar',$data);
        $this->load->view('staffpengendalian/daftar_siup'
,$data);
    }else{
        $id_status = "4";
        $data ['de'] = $this->staff->getcetak_siup($id_status);
        $data ['query']=$this->staff->getdatasiup($id_status);
        $this->load->view('backoffice/sidebar',$data);
        $this->load->view('staffdinas/daftar_siup',$data);
    }
}
```

5.2.4.3 Implementasi Kode Program Update Biodata

Pada Tabel 5.9 menunjukkan implementasi dari proses update biodata yang ada pada klas *controller* *pemohonfix_c* dengan mengambil salah satu method *biodata*.

Tabel 5.9 Kode Program Biodata

```
$username = $this->session->userdata('username');
    $data ['pengguna']=$this->pemohon->getnik($username);
    $data ['query']=$this->pemohon-
>getdataBio($username);
    $data ['pemberitahuan'] = $this->pemohon-
>getPemberitahuan($username);
```

```
$this->load->view('pemohon/sidebar_pemohon',$data);
$this->load->view('pemohon/tampil_bio',$data);
```

5.2.4.4 Implementasi Kode Program Menghapus Pemohon

Pada Tabel 5.10 merupakan kode program dari method lihatpemohon pada Klas *controller* staff_c yang merepresentasikan salah satu method dari proses hapus pemohon.

Tabel 5.10 Kode Program Lihatpemohon

```
$username = $this->session->userdata('username');
$data ['user']= $this->staff->getusername($username);
$data ['query']=$this->staff->getdataPemohon();

if($user == 'pengendalian'){
    $this->load->view('backoffice/sidebar',$data);
    $this->load->view('staffpengendalian/daftar_pemohon',
    $data);
}else if($user == 'kepalabidang'){
    $this->load->view('backoffice/sidebar',$data);
    $this->load->view('staffbidang/lihatpemohon',$data);
}else if($user == 'kepaladinas'){
    $this->load->view('backoffice/sidebar',$data);
    $this->load->view('staffbidang/lihatpemohon',$data);
}else{
    $this->load->view('backoffice/sidebar',$data);
    $this->load->view('staffpengendalian/no_akses',$data);
}
```

5.2.5 Implementasi Antarmuka

Implementasi antarmuka didasarkan pada perancangan sistem sebelumnya. Berikut merupakan antarmuka yang ada pada sistem perizinan.

5.2.5.1 Implementasi Halaman Pengajuan Perizinan SIUP

Pada Gambar 5.15 sampai dengan Gambar 5.18 merupakan implementasi antarmuka dari formulir pengajuan perizinan SIUP. Formulir perizinan SIUP terdiri dari kolom data dan upload berkas persyaratan perizinan. Kolom data berisi identitas diri dan identitas perusahaan.

Gambar 5.15 Halaman Form Pengajuan SIUP (1)

DPMPPTSP

Hi hinduniva

Halaman Awal

Data Pemohon

Izin SIUP

Izin TDP

Izin TDUP

Izin Reklame

Kelembagaan

Kegiatan Usaha

KBLI 4 digit

Barang / Jasa Dagangan Utama

Akte pendirian/ perubahan perusahaan berbentuk PT/ Koperasi(Badan Hukum)

Choose File No file chosen

Surat Keputusan Pengesahan Perusahaan Berbentuk PT/ Koperasi

Choose File No file chosen

Akte Pendirian Perusahaan yang telah didaftarkan di PN/ Perusahaan diluar PT

Choose File No file chosen

Gambar 5.16 Halaman Form Pengajuan SIUP (2)

DPMPPTSP

Hi hinduniva

Halaman Awal

Data Pemohon

Izin SIUP

Izin TDP

Izin TDUP

Izin Reklame

KK / KTP

Choose File No file chosen

Keterangan Domisili Tempat Usaha dari Kelurahan

Choose File No file chosen

NPWP / NPWP

Choose File No file chosen

Sertifikat Tanah

Choose File No file chosen

Neraca Awal Perusahaan

Choose File No file chosen

IMB

Choose File No file chosen

SPPL/ UKL/ Dokumen Ijin Lingkungan Lainnya

Choose File No file chosen

SIUP Lama Asli (untuk perpanjangan)

Choose File No file chosen

Pas Photo Background Berwarna

Choose File No file chosen

Laporan Kegiatan

Choose File No file chosen

Gambar 5.17 Halaman Form Pengajuan SIUP (3)

DPMPPTSP

Hi hinduniva

Kartu Kepesertaan BPJS/ Bukti biaya Kepesertaan

Choose File No file chosen

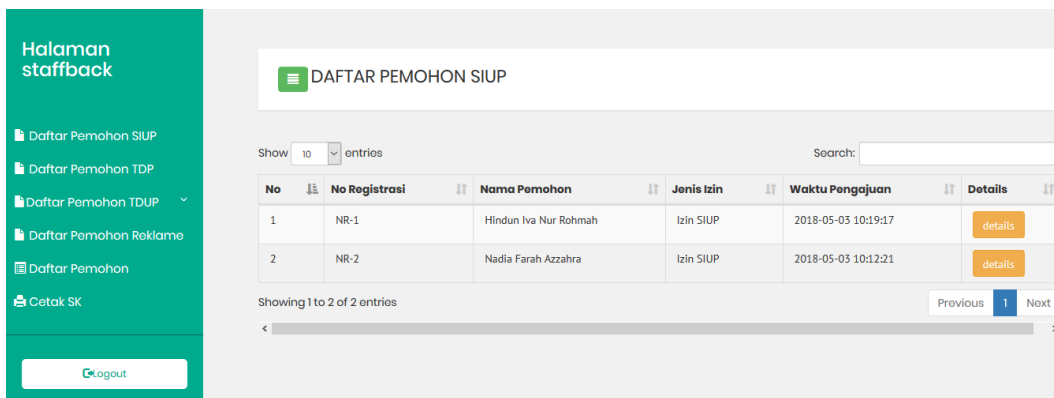
☐ Saya menyetujui segala ketentuan dan kebijakan yang berlaku.

Submit

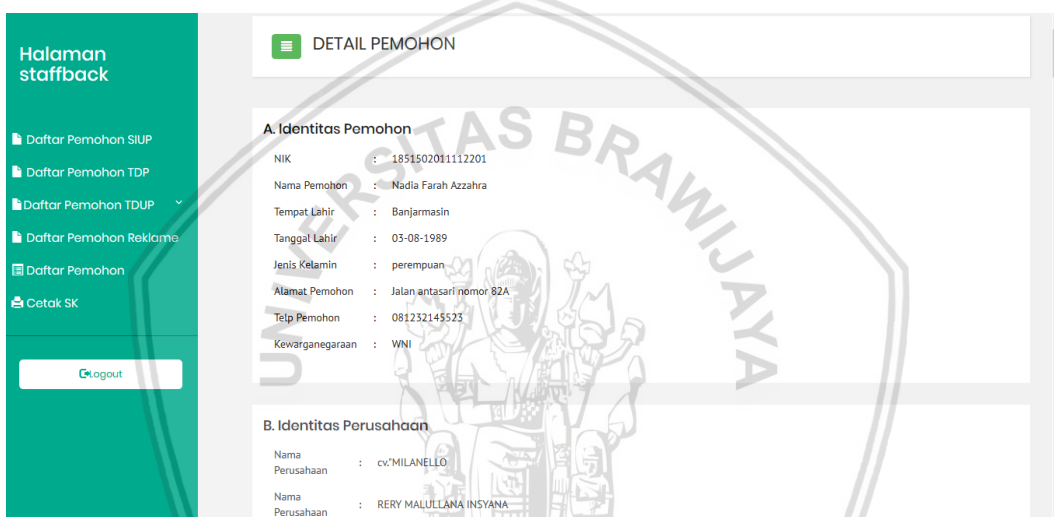
Gambar 5.18 Halaman Form Pengajuan SIUP (4)

5.2.5.2 Implementasi Halaman Validasi Perizinan

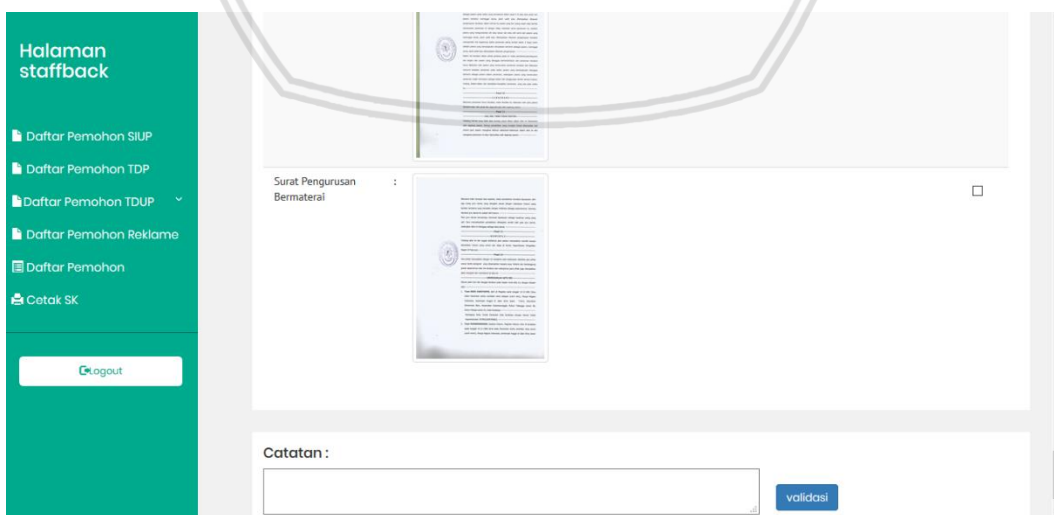
Pada Gambar 5.19 menampilkan halaman daftar pemohon SIUP. Pada Gambar 5.20 dan Gambar 5.21 merupakan implementasi dari halaman validasi perizinan yang dapat diakses oleh staff *back office*. Halaman validasi perizinan terdiri dari data detail pemohon SIUP yang terdiri dari identitas pemohon, identitas perusahaan, legalitas perusahaan dan upload berkas persyaratan perizinan. Sedangkan Gambar 5.22 merupakan tampilan dari form cetak yang akan ditampilkan jika pengajuan izin berhasil divalidasi.



Gambar 5.19 Halaman Daftar Pemohon SIUP



Gambar 5.20 Halaman Detail Pemohon (1)



Gambar 5.21 Halaman Detail Pemohon (2)

Halaman

- Daftar Pemohon SIUP
- Daftar Pemohon TDP
- Daftar Pemohon TDUP
- Daftar Pemohon Reklame
- Daftar Pemohon
- Cetak SK

Logout

FORM SURAT KEPUTUSAN SIUP

Form Surat Izin Usaha Perdagangan

Nomor: 503 / R / 423.107/2017

Nama: RERY MALULLANA INSYANA

Penanggungjawab: CV. MILANELLO

Nama Perusahaan: CV. MILANELLO

Alamat Perusahaan: JL. ABABIL NO.13 RT.003 RW.VIKEL TEMBOKREJO KEC. PURWOREJO KOTA PASURUAJ

Nomor Telp: (0343) 450 5905

Kekayaan Bersih: Rp. 500.000.000,- LIMA RATUS JUTA RUPIAH

Kelembagaan: KANTOR

Kegiatan Usaha: PERDAGANGAN BARANG DAN JASA (3314, 4530, 4540, 4641, 4649, 4651, 4779, 771)

Barang dan Jasa Dagangan: - JASA REPARASI PERALATAN LISTRIK- PERDAGANGAN BESAR SUKU CADANG DAN

Submit

Gambar 5.22 Halaman Form Cetak SIUP

5.2.5.3 Implementasi Halaman Update Biodata

Pada Gambar 5.23 merupakan implementasi halaman update biodata yang dapat digunakan oleh pemohon untuk memperbarui identitasnya. Formulir *update* biodata ini akan ditampilkan pada halaman awal pertama kali *login* jika pemohon belum memperbarui identitasnya.

DPMPSTP

Hi Farah Amalia

Sistem Perizinan / Data Pemohon / Tambah Data

Tambah Data Pemohon

No NIK: 148150201112201

Nama Lengkap: Nama Pemohon

Jenis Kelamin: ☐ Laki-laki ☐ Perempuan

Tempat Lahir: Tempat Lahir

Tanggal lahir: Tanggal lahir

Alamat: Alamat

Nomor Telp: Nomor Telp/ Handphone

Pekerjaan: PNS

Kewarganegaraan: Warga Negara Indonesia

Update Profile: No file chosen (max. 2MB)

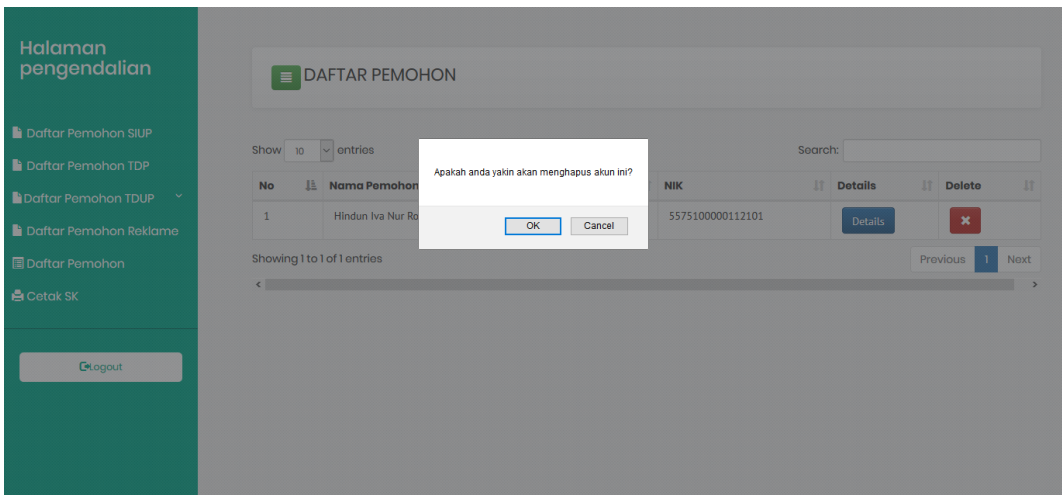
☐ Saya menyetujui segala ketentuan dan Kebijakan yang berlaku.

Submit

Gambar 5.23 Halaman Update Biodata

5.2.5.4 Implementasi Halaman Hapus Pemohon

Pada Gambar 5.24 merupakan implementasi dari halaman hapus pemohon yang diakses oleh kepala pengendalian. Pesan konfirmasi akan ditampilkan jika kepala pengendalian menekan tombol hapus. Apabila kepala pengendalian menekan tombol OK pada pesan konfirmasi maka semua data pemohon sampai dengan data perizinan yang diajukan pemohon tersebut akan otomatis terhapus juga.



Gambar 5.24 Halaman Hapus Pemohon













BAB 6 PENGUJIAN

Pada Bab ini menjelaskan mengenai pengujian yang akan dilakukan pada Sistem Perizinan SIUP, TDP, TDUP dan Reklame. Pengujian yang dilakukan adalah pengujian unit, pengujian integrasi dan pengujian validasi. Pada setiap pengujian disertakan juga analisis hasil pengujian yang bertujuan untuk memastikan bahwa kebutuhan yang telah didefinisikan pada tahap analisis kebutuhan dapat dijalankan oleh sistem.

6.1 Pengujian Unit

Pengujian unit dalam penelitian ini menggunakan teknik *basis path testing* untuk menguji setiap operasi dari setiap klas berdasarkan dengan algoritme yang dihasilkan pada tahap perancangan. Algoritme yang diuji antara lain `action_siup()`, `verifikasi_siup()`, `form_bio()` dan `hapus_pemohon()`. Proses pengujian unit dilakukan dengan membentuk *node* dari algoritme yang dihasilkan pada tahap perancangan, dilanjutkan dengan memodelkan *flowgraph*. Setelah didapatkan gambaran *flowgraph* maka mencari jumlah kompleksitas siklomatis (*cyclomatic complexity*) dengan menggunakan persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis. E atau *edge* merupakan penghubung antar node dan N merupakan jumlah simpul node. Dilanjutkan dengan perhitungan $V(G) = P + 1$, dimana p merupakan jumlah *predicate node* atau bisa disebut *node* yang memiliki cabang. Kemudian menentukan jalur independen dan memberikan kasus uji.

6.1.1 Pengujian Unit Action_siup

Pada Gambar 6.1 digambarkan mengenai pembentukan *node* algoritme dari method `action_siup` dan dilanjutkan dengan pembentukan *flowgraph* yang digambarkan pada Gambar 6.2 yang berfungsi menggambarkan alur dari method `action_siup`.

a. Algoritme method `action_siup`

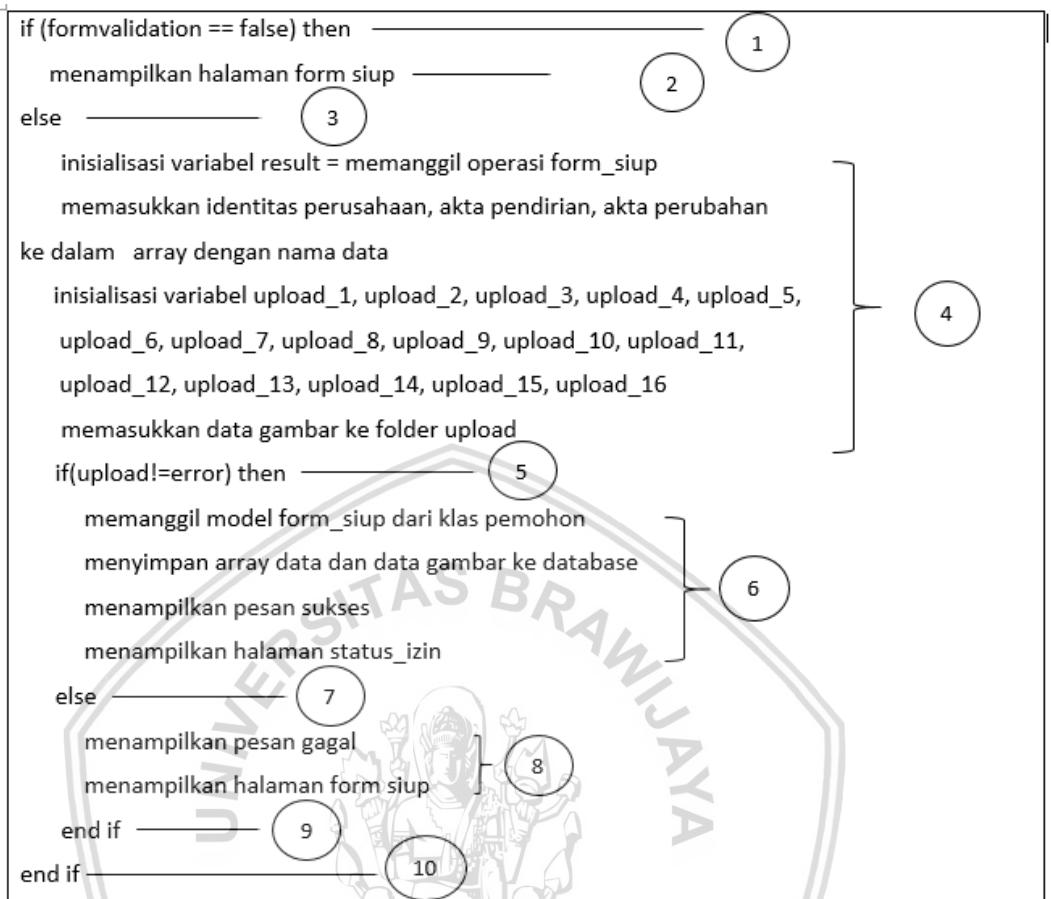
Berdasarkan *flowgraph* dari Gambar 6.2 maka dilakukan perhitungan *cyclomatic complexity*.

- $V(G) = E - N + 2$
- $V(G) = 11 - 10 + 2$
- $V(G) = 3$
- $V(G) = P + 1 = 2 + 1 = 3$

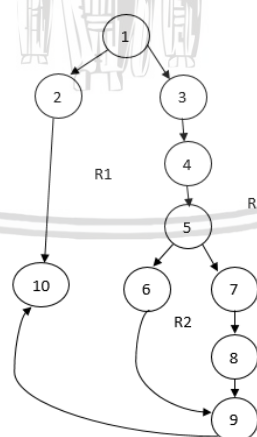
Berdasarkan hasil nilai dari *cyclomatic complexity* didapatkan 3 buah jalur independen yaitu:

- Jalur 1: 1 – 2 – 10
- Jalur 2: 1 – 3 – 4 – 5 – 6 – 9 – 10

- Jalur 3: 1 – 3 – 4 – 5 – 7 – 8 – 9 – 10



Gambar 6.1 Pembentukan Node Algoritme Action_siup



Gambar 6.2 Flowgraph Action_siup

Berdasarkan 3 jalur independen yang didapatkan, maka dilakukan pengujian terhadap tiap jalur tersebut. Pada Tabel 6.1 merupakan pemaparan kasus uji dari algoritme action_siup ().

Tabel 6.1 Hasil Pengujian Unit dari Method Action_siup

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1.	1	Memanggil operasi action_siup dengan form siup yang berisi data kosong	Akan menampilkan halaman form siup dengan pesan kesalahan yang terdapat pada tiap kolom	Menampilkan halaman form siup dengan pesan kesalahan yang terdapat pada tiap kolom	Valid
2.	2	Memanggil operasi action_siup dengan variabel identitas perusahaan, akta pendirian, akta perubahan dan upload data gambar berkas perizinan	Menyimpan data ke database dan menampilkan halaman status_izin	Menyimpan data ke database dan menampilkan halaman status_izin	Valid
3.	3	Memanggil operasi action_siup dengan variabel identitas perusahaan, akta pendirian, akta perubahan tanpa mengupload gambar berkas persyaratan perizinan.	Menampilkan pesan “Gagal! Gambar belum di upload, Upload Gambar Terlebih Dahulu!” pada form siup	Menampilkan pesan “Gagal! Gambar belum di upload, Upload Gambar Terlebih Dahulu!” pada form siup	Valid

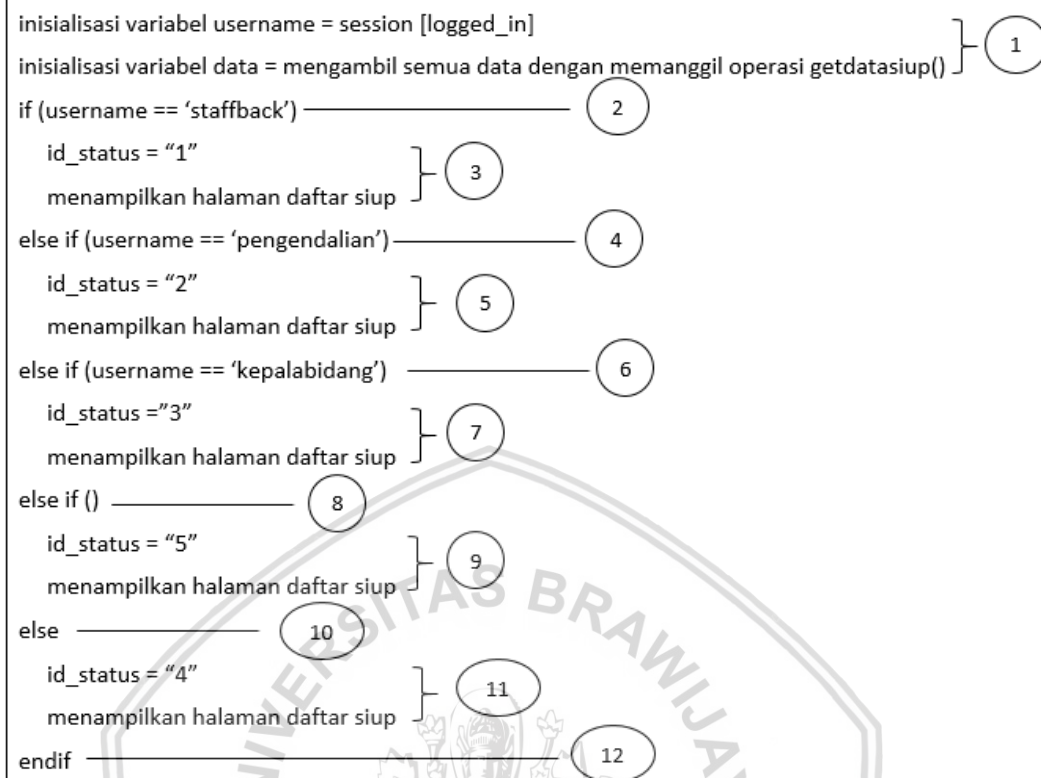
b. Analisis Hasil Pengujian

Hasil pengujian pada method action_siup bernilai valid pada setiap kasus uji yang dilakukan. Pengujian ini telah sesuai dengan hasil yang diharapkan.

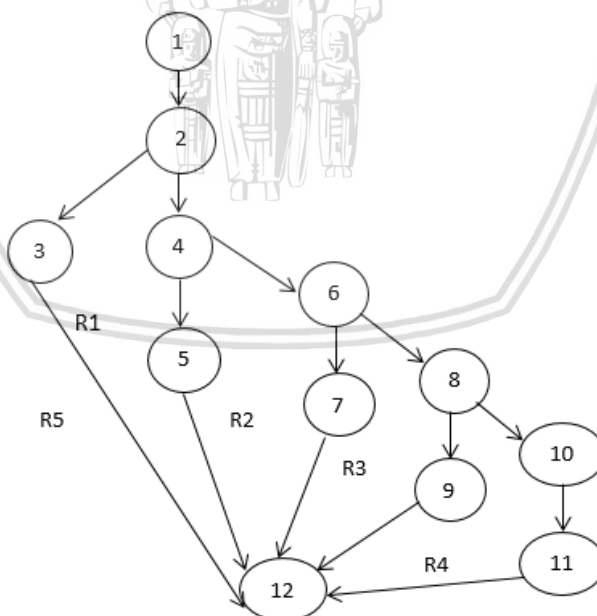
6.1.2 Pengujian Unit Daftar_siup

Pada Gambar 6.3 dijelaskan mengenai pembentukan *node* algoritme method daftar_siup yang berasal dari klas staff_c dan gambaran *flowgraph* dari algoritme daftar_siup yang ditunjukkan pada Gambar 6.4

a. Algoritme pada method daftar_siup



Gambar 6.3 Pembentukan Node Algoritme Daftar_siup



Gambar 6.4 Flowgraph Daftar_siup

Berdasarkan *flowgraph* yang dihasilkan pada Gambar 6.4 maka dapat dilakukan perhitungan *cyclomatic complexity*.

- $V(G) = \text{Jumlah Region} = 5$

- $V(G) = E - N + 2 = 15 - 12 + 2 = 5$
- $V(G) = P + 1 = 4 + 1 = 5$

Berdasarkan hasil dari perhitungan *cyclomatic complexity* didapatkan 2 jalur independen yaitu:

- Jalur 1: 1 – 2 – 3 – 12
- Jalur 2: 1 – 2 – 4 – 5 – 12
- Jalur 3: 1 – 2 – 4 – 6 – 7 – 12
- Jalur 4: 1 – 2 – 4 – 6 – 8 – 9 – 12
- Jalur 5: 1 – 2 – 4 – 6 – 8 – 10 – 11 – 12

Berdasarkan 5 jalur independen yang didapatkan, maka dilakukan pengujian terhadap tiap jalur tersebut. Pada Tabel 6.2 merupakan pemaparan kasus uji dari algoritme daftar_siup ().

Tabel 6.2 Hasil Pengujian Unit dari Method Daftar_siup

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1.	1	Memanggil fungsi daftar_siup dengan username = staffback	Menampilkan halaman daftar siup dengan id_status = 1	Menampilkan halaman daftar siup dengan id_status= 1	Valid
2.	2	Memanggil fungsi daftar_siup dengan username = pengendalian	Menampilkan halaman daftar siup dengan id_status = 2	Menampilkan halaman daftar siup dengan id_status = 2	Valid
3.	3	Memanggil fungsi daftar_siup dengan username = kepalabidang	Menampilkan halaman daftar siup dengan id_status = 3	Menampilkan halaman daftar siup dengan id_status = 3	Valid
4.	4	Memanggil fungsi daftar_siup dengan username = kepalaadmin	Menampilkan halaman daftar siup dengan id_status = 5	Menampilkan halaman daftar siup dengan id_status = 5	Valid
5.	5	Memanggil fungsi daftar_siup dengan username = kepaladinas	Menampilkan daftar siup dengan id_status = 4	Menampilkan daftar siup dengan id_status = 4	Valid

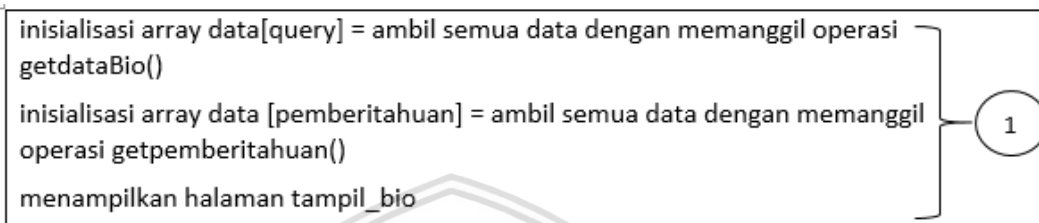
b. Analisis Hasil

Berdasarkan hasil dari pengujian yang dilakukan maka semua kasus uji bernilai valid sesuai dengan hasil yang diharapkan.

6.1.3 Pengujian Unit Biodata

Pada pengujian unit update biodata method yang diuji adalah method `form_bio` yang berasal dari klas `pemohonfix_c`. Gambar 6.5 menunjukkan pembentukan dari *node* untuk method `form_bio` dilanjutkan dengan menggambarkan *flowgraph* dari method tersebut yang ditunjukkan pada Gambar 6.6

a. Algoritme pada method `form_bio`



Gambar 6.5 Pembentukan Node Algoritme Biodata

1

Gambar 6.6 Flowgraph Biodata

Berdasarkan *flowgraph* yang dihasilkan pada Gambar 6.6 maka dapat dilakukan perhitungan *cyclomatic complexity*.

- $V(G) = E - N + 2$
- $V(G) = 0 - 1 + 2 = 1$
- $V(G) = P + 1 = 0 + 1 = 1$

Dari hasil perhitungan *cyclomatic complexity* maka didapatkan 3 buah jalur independen yaitu:

- Jalur 1: 1

Berdasarkan 1 jalur independen yang didapatkan, maka dilakukan pengujian terhadap tiap jalur tersebut. Pada Tabel 6.3 merupakan pemaparan kasus uji dari algoritme biodata ().

Tabel 6.3 Hasil Pengujian Unit Method Biodata

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1.	1	Memanggil fungsi biodata	Menampilkan halaman biodata pemohon	Menampilkan halaman biodata pemohon	Valid

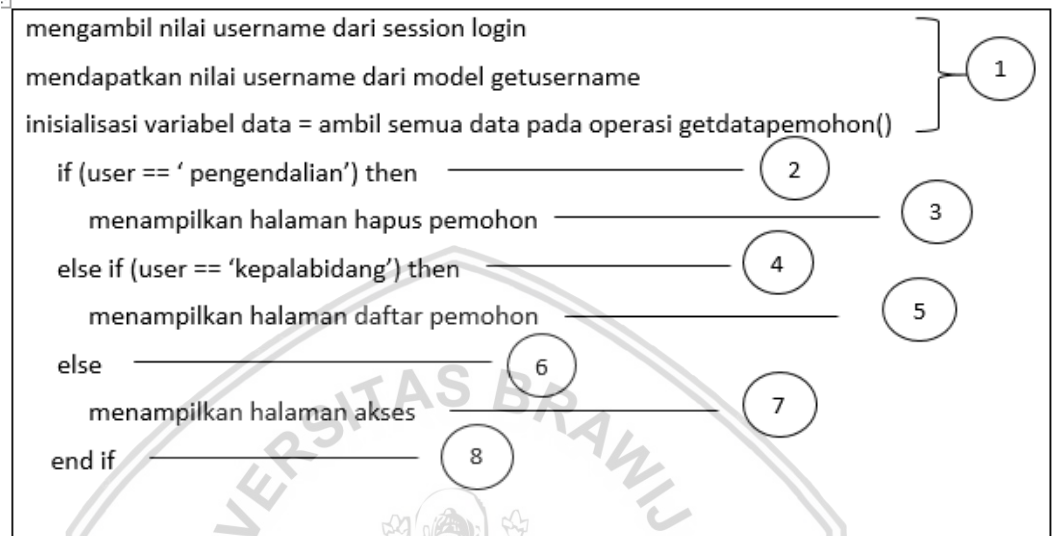
b. Analisis Hasil

Berdasarkan dari hasil pengujian yang telah dijabarkan pada Tabel 6.3 dimana semua kasus uji bernilai valid dan sesuai dengan hasil yang diharapkan.

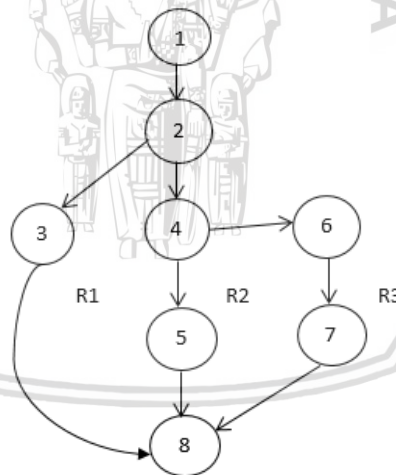
6.1.4 Pengujian Unit Lihatpemohon

Pada Gambar 6.7 menunjukkan pembentukan dari *node* method lihatpemohon yang berasal dari klas pemohonfix_c dan menggambarkan *flowgraph* dari fungsi tersebut yang ditunjukkan pada Gambar 6.8

a. Algoritme pada method lihatpemohon



Gambar 6.7 Pembentukan Node Method Lihatpemohon



Gambar 6.8 *Flowgraph* Lihatpemohon

Berdasarkan hasil dari *flowgraph* method lihatpemohon pada Gambar 6.8 maka dapat dilakukan perhitungan nilai *cyclometric complexity*.

- $V(G) = E - N + 2$
- $V(G) = 9 - 8 + 2 = 3$
- $V(G) = P + 1 = 2 + 1 = 3$

Dari hasil perhitungan *cyclometric complexity* di dapatkan 3 jalur independen yaitu

- Jalur 1: 1 – 2 – 3 – 8
- Jalur 2: 1 – 2 – 4 – 5 – 8
- Jalur 3: 1 – 2 – 4 – 6 – 7 – 8

Berdasarkan 3 jalur independen yang didapatkan, maka dilakukan pengujian terhadap tiap jalur tersebut. Pada Tabel 6.4 merupakan pemaparan kasus uji dari algoritme lihatpemohon ().

Tabel 6.4 Hasil Pengujian Unit Method Lihatpemohon

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1.	1	Memanggil fungsi lihatpemohon dengan nilai <i>username</i> ="pengendalian".	Menampilkan halaman hapuspemohon.	Menampilkan halaman hapuspemohon.	Valid
2.	2	Memanggil fungsi lihatpemohon dengan nilai <i>username</i> ="kepalabidang"	Menampilkan halaman lihat pemohon.	Menampilkan halaman lihat pemohon	Valid
3.	3	Memanggil fungsi lihatpemohon dengan nilai <i>username</i> ="staffback".	Menampilkan halaman no_akses.	Menampilkan halaman no_akses.	Valid

b. Analisis hasil

Berdasarkan dari hasil pengujian yang telah dijabarkan pada Tabel 6.4 dimana semua kasus uji bernilai valid dan sesuai dengan hasil yang diharapkan.

6.2 Pengujian Integrasi

Pada pengujian ini akan menguji mengenai keterkaitan atau hubungan antar method ataupun klas. Operasi yang akan diuji yaitu operasi *update_siup* pada klas *staff_c*, *validasi_siup* pada klas *staff_c*, *form_bio* pada klas *pemohonfix_c* dan *index* pada klas *staff_c*.

6.2.1 Pengujian Integrasi Update_siup

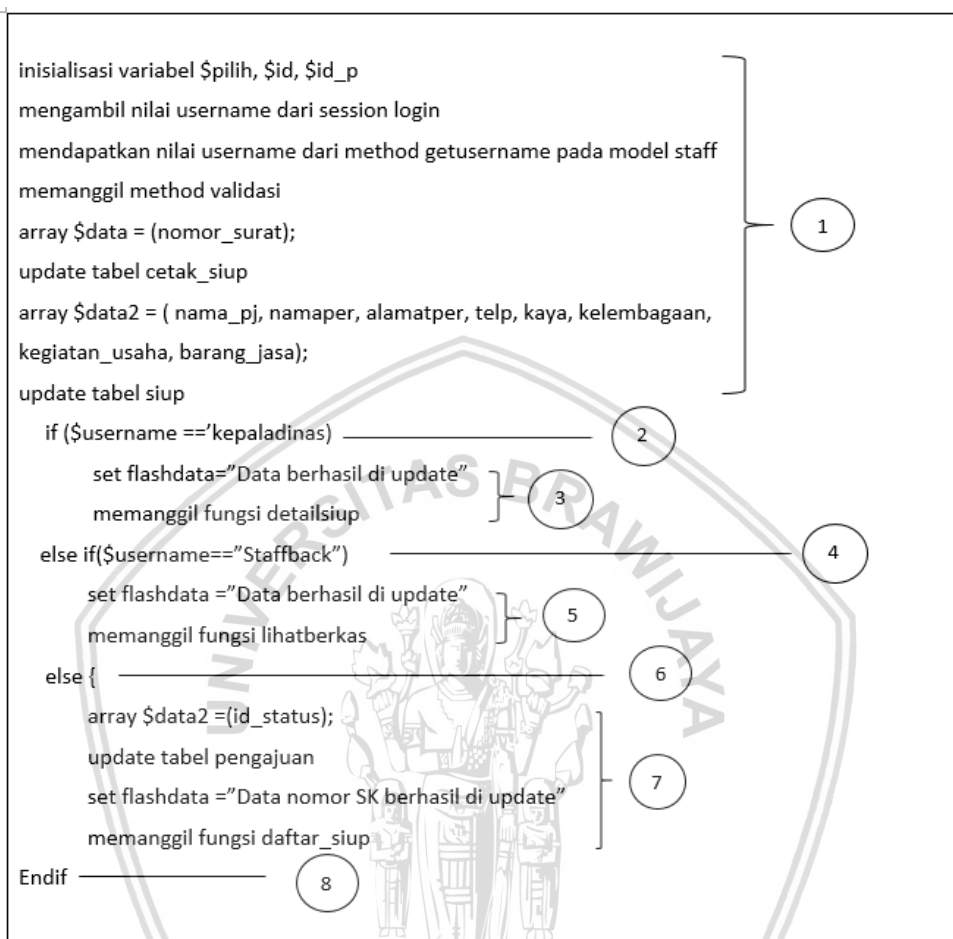
Pada method *update_siup* pada klas *staff_c* akan memanggil method *detailsiup()*, *lihatberkas()* dan *daftar_siup()* yang ada pada klas *staff_c*. Algoritme dan pembentukan *node* dapat dilihat pada Gambar 6.9 sedangkan Gambar 6.10 menunjukkan *flowgraph* dari method *update_siup*.

a. Algoritme method *update_siup()*

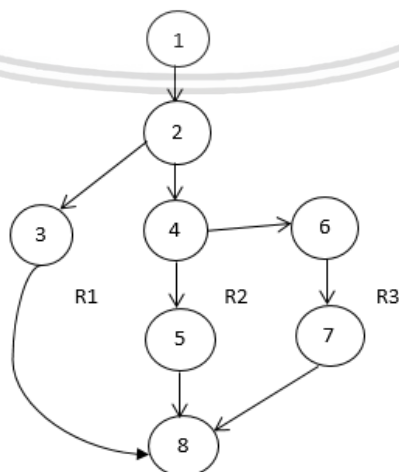
Berdasarkan gambaran dari *flowgraph* yang ditunjukkan pada Gambar 6.10 maka dapat dilakukan perhitungan *cyclomatic complexity* sebagai berikut:

$$- V(G) = E - N + 2$$

- $V(G) = 12 - 11 + 2 = 3$
- $V(G) = 2 + 1 = 2 + 1 = 3$



Gambar 6.9 Pembentukan *Node* Algoritme Update_siup



Gambar 6.10 *Flowgraph* Method Update_siup

Dari hasil perhitungan *cyclomatic complexity* maka didapatkan 3 jalur independen yaitu:

- Jalur 1: 1 – 2 – 3 – 8
- Jalur 2: 1 – 2 – 4 – 5 – 8
- Jalur 3: 1 – 2 – 4 – 6 – 7 – 8

Berdasarkan 3 jalur independen yang didapatkan, maka dilakukan pengujian terhadap jalur tersebut. Tabel 6.5 merupakan pemaparan kasus uji dari algoritme method `update_siup` ().

Tabel 6.5 Hasil Pengujian Integrasi Method `Update_siup`

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1.	1	Memanggil fungsi <code>update_siup</code> dengan nilai variabel <code>\$username='kepaladinas'</code>	Update nomor surat pada tabel <code>cetak_siup</code> dan memperbarui data sk pada tabel <code>siup</code> . Sistem akan menampilkan halaman detail <code>siup</code> .	Update nomor surat pada tabel <code>cetak_siup</code> dan memperbarui data sk pada tabel <code>siup</code> . Sistem akan menampilkan halaman detail <code>siup</code> .	Valid
2.	2	Memanggil fungsi <code>update_siup</code> dengan nilai variabel <code>\$username='staffback'</code>	Update nomor surat pada tabel <code>cetak_siup</code> dan memperbarui data sk pada tabel <code>siup</code> . Sistem akan menampilkan halaman lihat berkas.	Update nomor surat pada tabel <code>cetak_siup</code> dan memperbarui data sk pada tabel <code>siup</code> . Sistem akan menampilkan halaman lihat berkas.	Valid
3.	3	Memanggil fungsi <code>update_siup</code> dengan nilai variabel <code>\$username='kepalaadmin'</code>	Update nomor surat pada tabel <code>cetak_siup</code> , memperbarui data sk pada tabel <code>siup</code> dan memperbarui <code>id_status</code> pada tabel <code>pengajuan</code> . Sistem akan	Update nomor surat pada tabel <code>cetak_siup</code> , memperbarui data sk pada tabel <code>siup</code> dan memperbarui <code>id_status</code> pada tabel <code>pengajuan</code> .	Valid

			menampilkan halaman daftar siup.	Sistem akan menampilkan halaman daftar siup.	
--	--	--	----------------------------------	--	--

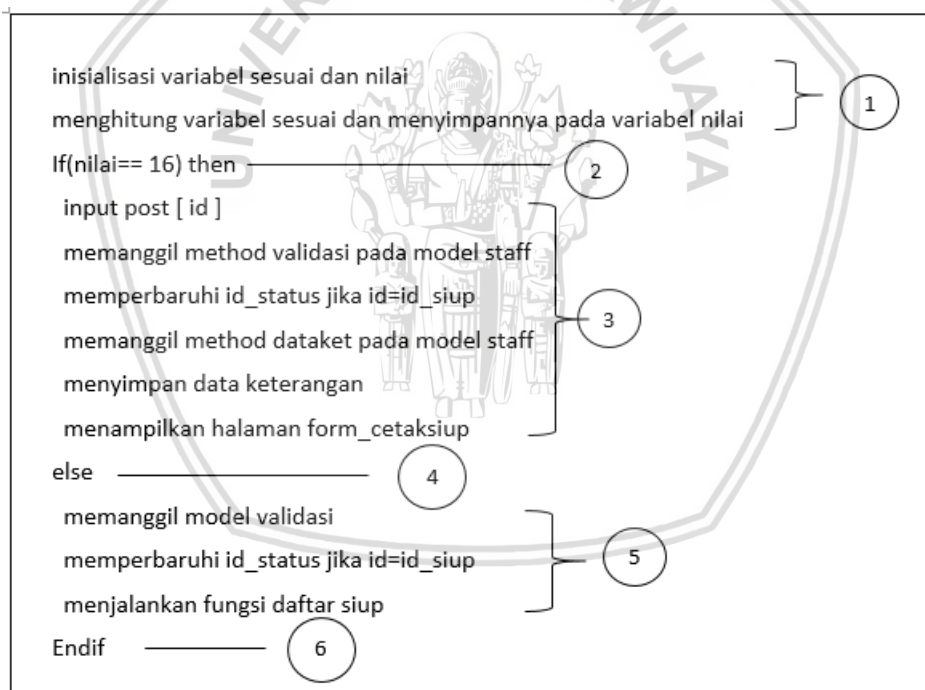
b. Analisis hasil

Hasil pengujian pada method status_izin bernilai valid pada setiap kasus uji yang dilakukan. Pengujian ini telah sesuai dengan hasil yang diharapkan.

6.2.2 Pengujian Integrasi Validasi_siup

Pada method validasi_siup yang ada di klas staff_c akan memanggil method validasi() pada klas staff, memanggil method dataket() pada klas staff dan memanggil method daftar_siup() pada klas staff_c. Gambar 6.11 akan menjelaskan algoritme dan pembentukan *node* dari method validasi_siup dan Gambar 6.12 menggambarkan *flowgraph* dari method tersebut.

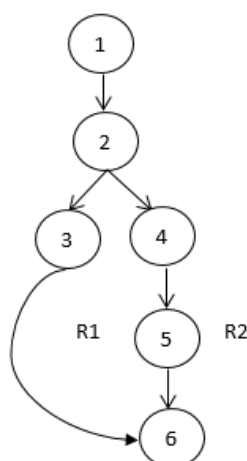
a. Algoritme operasi validasi_siup



Gambar 6.11 Pembentukan Node Validasi_siup

Berdasarkan dari hasil *flowgraph* yang digambarkan pada Gambar 6.12 maka dapat dilakukan perhitungan *cyclomatic complexity* sebagai berikut:

- $V(G) = \text{Jumlah Region} = 2$
- $V(G) = E - N + 2 = 8 - 8 + 2 = 2$
- $V(G) = P + 1 = 1 + 1 = 2$



Gambar 6.12 Flowgraph Validasi_siup

Berdasarkan hasil dari perhitungan *cyclomatic complexity* didapatkan 2 jalur independen yaitu:

- Jalur 1: 1 – 2 – 3 – 6
- Jalur 2: 1 – 2 – 4 – 5 – 6

Berdasarkan 2 jalur independen yang didapatkan, maka dilakukan pengujian terhadap tiap jalur tersebut. Pada Tabel 6.6 merupakan pemaparan kasus uji dari algoritme validasi_siup ().

Tabel 6.6 Hasil Pengujian Integrasi Method Validasi_siup

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1.	1	Memanggil fungsi verifikasi_siup dengan nilai =16	Memperbarui id_status pada tabel siup, memasukkan data keterangan pada tabel keterangan_siup di <i>database</i> dan menampilkan halaman form cetaksiup	Memperbarui id_status pada tabel siup dan memasukkan data keterangan pada <i>database</i> dan menampilkan halaman form cetak siup	Valid
1.	2	Memanggil fungsi validasi_siup dengan nilai = 1	Memperbarui id_status pada tabel siup dengan nilai 1 dan menampilkan	Memperbarui id_status pada tabel siup dengan nilai 1 dan menampilkan	Valid

			halaman daftar siup	halaman daftar siup	
--	--	--	------------------------	------------------------	--

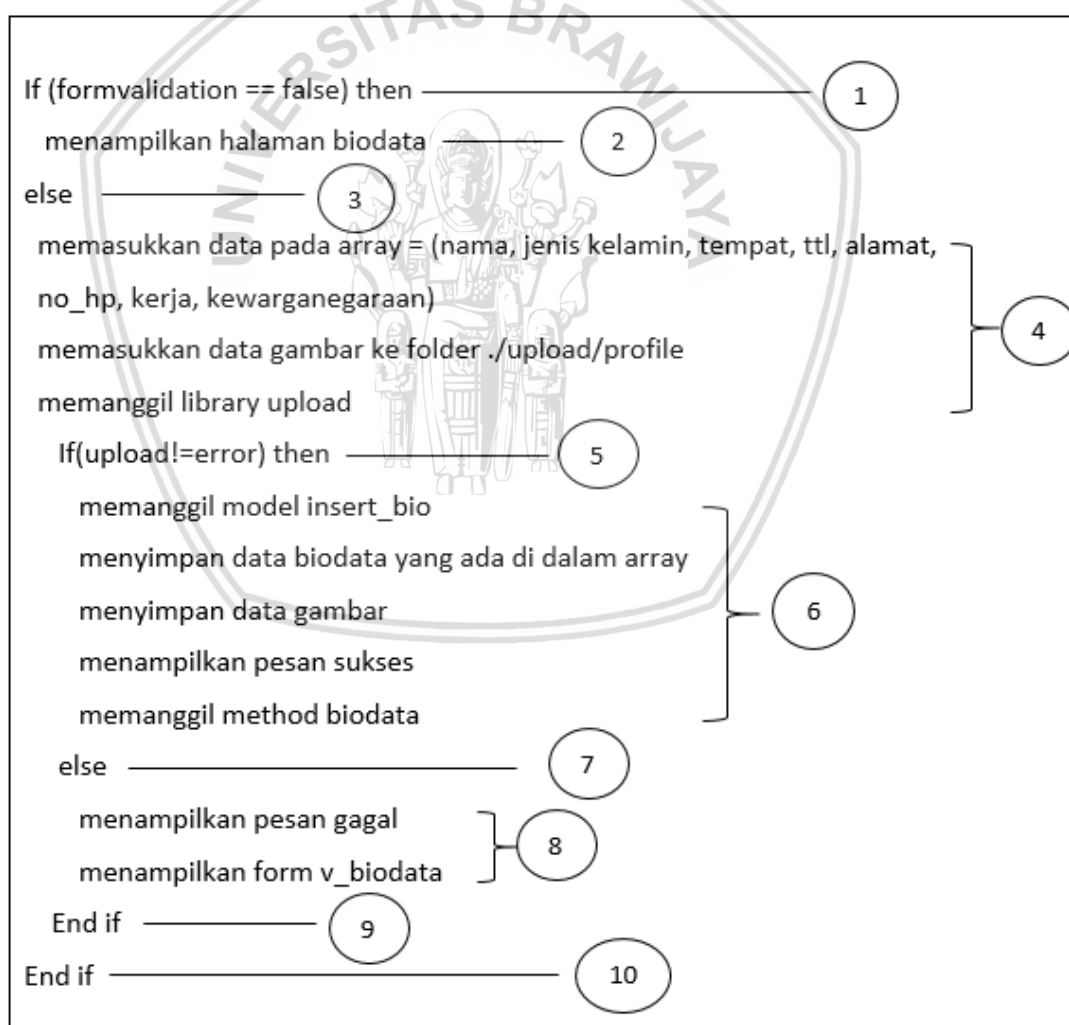
b. Analisis Hasil

Berdasarkan hasil dari pengujian yang dilakukan maka semua kasus uji bernilai valid sesuai dengan hasil yang diharapkan.

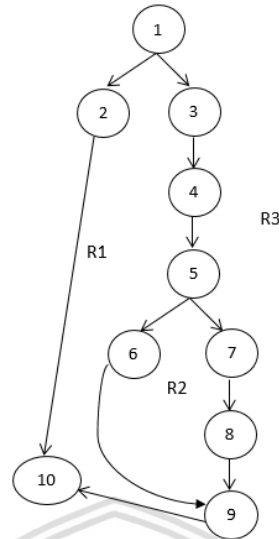
6.2.3 Pengujian Integrasi Method Form_bio

Pada method form_bio yang terdapat pada klas pemohonfix_c akan memanggil library *upload* pada klas image_upload, model insert_bio() dan memanggil method biodata(). Pada Gambar 6.13 akan menjelaskan mengenai algoritme dan pembentukan *node* dari method form_bio. Gambar 6.14 menggambarkan *flowgraph* dari algoritme method form_bio.

a. Algoritme operasi form_bio



Gambar 6.13 Pembentukan *Node* Method Form_bio



Gambar 6.14 Flowgraph Method Form_bio

Berdasarkan *flowgraph* yang dihasilkan pada Gambar 6.14 maka dapat dilakukan perhitungan cyclomatic complexity.

- $V(G) = E - N + 2$
- $V(G) = 11 - 10 + 2 = 3$
- $V(G) = P + 1 = 2 + 1 = 3$

Dari hasil perhitungan *cyclomatic complexity* maka didapatkan 3 buah jalur independen yaitu:

- Jalur 1: 1 – 2 – 10
- Jalur 2: 1 – 3 – 4 – 5 – 6 – 9 – 10
- Jalur 3: 1 – 3 – 4 – 5 – 7 – 8 – 9 – 10

Berdasarkan 3 jalur independen yang didapatkan, maka dilakukan pengujian terhadap tiap jalur tersebut. Pada Tabel 6.7 merupakan pemaparan kasus uji dari algoritme form_bio ().

Tabel 6.7 Hasil Pengujian Integrasi Method Form_bio

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1.	1	Memanggil fungsi form_bio dengan form bernilai data kosong	Menampilkan pesan kesalahan pada setiap kolom yang ada di halaman fom biodata	Menampilkan pesan kesalahan pada setiap kolom yang ada di halaman fom biodata	Valid
2.	2	Memanggil fungsi form_bio dengan nilai	Menyimpan variabel nama,	Menyimpan variabel nama,	Valid

		variabel nama="Hindun Iva Nur Rohmah", tempat lahir ="Pasuruan", ttl_bio ="29-11-1995", alamat ="jalan bunga cempaka 18 B", no_telp ="085790806155", kerja ="PNS", gender="perempuan", kewarganegaraan ="WNI" dan mengupload foto profil.	tempat lahir, ttl_bio, alamat, no_telp, kerja, gender, kewarganegaraan dalam tabel biodata_pemohon yang terdapat di database, menampilkan pesan "Update Biodata Berhasil" dan menampilkan halaman biodata.	tempat lahir, ttl_bio, alamat, no_telp, kerja, gender, kewarganegaraan dalam tabel biodata_pemohon yang terdapat di database, menampilkan pesan "Update Biodata Berhasil" dan menampilkan halaman biodata.	
3.	3	Memanggil fungsi form_bio dengan nilai variabel nama="Hindun Iva Nur Rohmah", tempat lahir ="Pasuruan", ttl_bio ="29-11-1995", alamat ="jalan bunga cempaka 18 B", no_telp ="085790806155", kerja ="PNS", gender="perempuan", kewarganegaraan ="WNI" tanpa upload foto profil	Menampilkan pesan "Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!" dan menampilkan halaman formulir biodata	Menampilkan pesan "Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!" dan menampilkan halaman formulir biodata	Valid

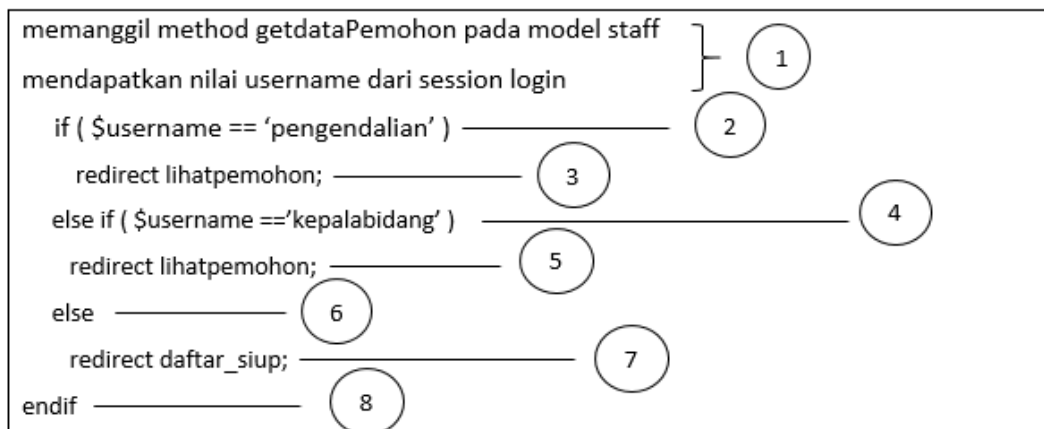
b. Analisis hasil

Hasil pengujian pada method upload bernilai valid pada setiap kasus uji yang dilakukan. Pengujian ini telah sesuai dengan hasil yang diharapkan.

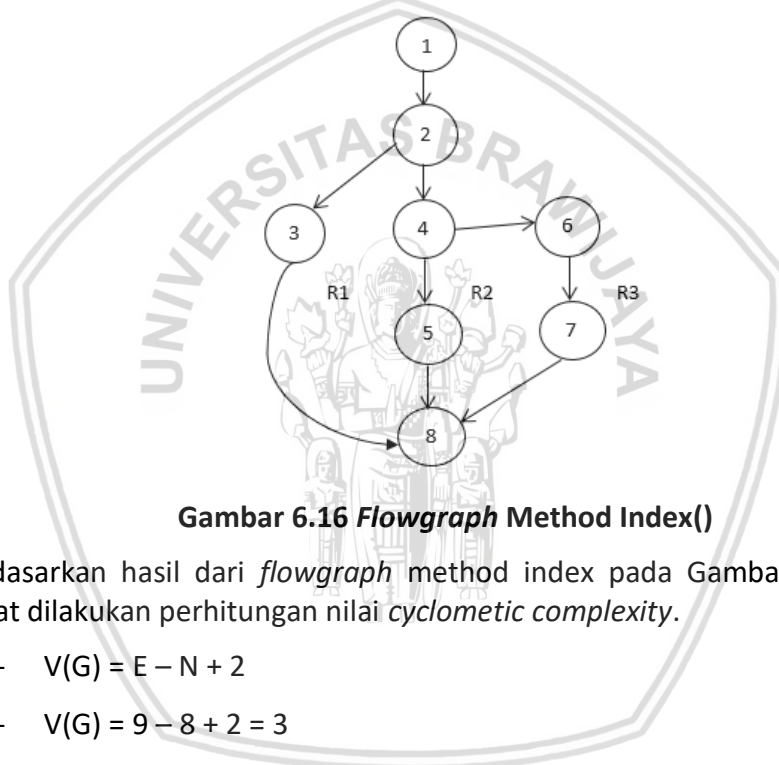
6.2.4 Pengujian Integrasi Method Index()

Pada method index pada klas staff_c akan memanggil method lihatpemohon dan method daftar_siup dari klas staff_c. Pada Gambar 6.15 akan dijelaskan mengenai algoritme dan pembentukan *node* dari method lihatpemohon. Gambar 6.16 menunjukkan *flowgraph* dari method index().

a. Algoritme pada method index()



Gambar 6.15 Pembentukan Node Method Index()



Gambar 6.16 Flowgraph Method Index()

Berdasarkan hasil dari *flowgraph* method index pada Gambar 6.16 maka dapat dilakukan perhitungan nilai *cyclometric complexity*.

- $V(G) = E - N + 2$
- $V(G) = 9 - 8 + 2 = 3$
- $V(G) = P + 1 = 2 + 1 = 3$

Dari hasil perhitungan *cyclometric complexity* di dapatkan 3 jalur independen yaitu

- Jalur 1: 1 – 2 – 3 – 8
- Jalur 2: 1 – 2 – 4 – 5 – 8
- Jalur 3: 1 – 2 – 4 – 6 – 7 – 8

Berdasarkan 3 jalur independen yang didapatkan, maka dilakukan pengujian terhadap tiap jalur tersebut. Pada Tabel 6.8 merupakan pemaparan kasus uji dari algoritme lihatpemohon ().

Tabel 6.8 Hasil Pengujian Integrasi Method Lihatpemohon

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1.	1	Memanggil fungsi index dengan nilai <i>username</i> ="pengendalian".	Berhasil memanggil fungsi lihat pemohon	Berhasil memanggil fungsi lihat pemohon	Valid
2.	2	Memanggil fungsi index dengan nilai <i>username</i> ="kepalabidang"	Berhasil memanggil fungsi lihatpemohon	Berhasil memanggil fungsi lihatpemohon	Valid
3.	3	Memanggil fungsi index dengan nilai <i>username</i> ="staffback".	Berhasil memanggil fungsi daftar siup	Berhasil memanggil fungsi daftar siup	Valid

b. Analisis Hasil

Hasil pengujian pada method lihatpemohon bernilai valid pada setiap kasus uji yang dilakukan. Pengujian ini sesuai dengan hasil yang diharapkan.

6.3 Pengujian Validasi

Pengujian validasi digunakan untuk menguji semua kebutuhan fungsional sistem. Pengujian ini akan dilakukan dengan metode pengujian *blackbox* yaitu memeriksa semua fungsi-fungsi yang terdapat di dalam sistem, memastikan bahwa fungsi tersebut berjalan sesuai dengan perancangan kebutuhan fungsional yang telah didefinisikan dan tidak ada kesalahan. Proses pengujian validasi dilakukan berdasarkan dari hasil analisis kebutuhan yang mengacu pada skenario *use case*. Kode pengujian validasi menggunakan PV_001. PV menyakatan singkatan dari Pengujian Validasi dan 001 adalah nomor urut pengujian.

6.3.1 Pengujian Validasi Registrasi

a. Kasus uji registrasi dengan data lengkap

Tabel 6.9 Kasus Uji Registrasi dengan Data Lengkap

Kode Kasus Uji	PV_001
Nama Kasus Uji	Kasus uji registrasi dengan data lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat melakukan registrasi dengan data tidak kosong.
Prosedur Pengujian	1. Membuka formulir registrasi.

	2. Memasukkan <i>username</i> , <i>password</i> , <i>confirm password</i> , NIK dan email. 3. Menekan tombol daftar.
Hasil yang diharapkan	Sistem akan menampilkan halaman awal pemohon.
Hasil Pengujian	Sistem menampilkan halaman awal pemohon.
Status	Valid

- b. Kasus uji registrasi dengan data kosong

Tabel 6.10 Kasus Uji Registrasi dengan Data Kosong

Kode Kasus Uji	PV_002
Nama Kasus Uji	Kasus uji registrasi dengan data kosong
Tujuan Pengujian	Untuk memastikan bahwa jika registrasi dengan data kosong maka sistem akan menampilkan pesan kesalahan.
Prosedur Pengujian	1. Membuka formulir registrasi. 2. Menekan tombol daftar.
Hasil yang diharapkan	Sistem akan menampilkan pesan error pada setiap kolom berisi kolom wajib diisi!.
Hasil Pengujian	Sistem menampilkan pesan error pada setiap kolom yang berisi kolom wajib diisi!.
Status	Valid

6.3.2 Pengujian Validasi *Login*

- a. Kasus uji berhasil login sebagai pengguna

Tabel 6.11 Kasus Uji Berhasil *Login* Sebagai Pengguna

Kode Kasus Uji	PV_003
Nama Kasus Uji	Kasus uji berhasil login sebagai pengguna
Tujuan Pengujian	Untuk memastikan jika pengguna dapat melakukan <i>login</i> sesuai dengan <i>username</i> dan <i>password</i> nya.
Prosedur Pengujian	1. Membuka formulir <i>login</i> . 2. Memasukkan <i>username</i> dan <i>password</i> . 3. Menekan tombol masuk.

Hasil yang diharapkan	Sistem akan menampilkan halaman biodata pemohon.
Hasil Pengujian	Sistem menampilkan halaman biodata pemohon.
Status	Valid

Kasus uji *username* dan *password* kosong

Tabel 6.12 Kasus Uji *Username* dan *Password* Kosong

Kode Kasus Uji	PV_004
Nama Kasus Uji	Kasus uji <i>username</i> dan <i>password</i> kosong
Tujuan Pengujian	Untuk memastikan bahwa pengguna tidak dapat melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang kosong serta sistem menampilkan pesan peringatan.
Prosedur Pengujian	1. Membuka formulir <i>login</i> . 2. Menekan tombol masuk.
Hasil yang diharapkan	Sistem akan menampilkan pesan “kolom <i>username</i> wajib diisi, kolom <i>password</i> wajib diisi”.
Hasil Pengujian	Sistem menampilkan pesan “kolom <i>username</i> wajib diisi, kolom <i>password</i> wajib diisi”.
Status	Valid

c. Kasus uji *username* dan *password* tidak sesuai

Tabel 6.13 Kasus Uji *Username* dan *Password* Tidak Sesuai

Kode Kasus Uji	PV_005
Nama Kasus Uji	Kasus uji <i>username</i> dan <i>password</i> tidak sesuai
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika pengguna memasukkan data <i>username</i> dan <i>password</i> yang tidak sesuai.
Prosedur Pengujian	1. Membuka formulir <i>login</i> . 2. Memasukkan <i>username</i> dan <i>password</i> yang salah. 3. Menekan tombol masuk

Hasil yang diharapkan	Sistem akan menampilkan pesan “Gagal Login: Cek <i>username</i> dan <i>password</i> anda!”.
Hasil Pengujian	Sistem menampilkan pesan “Gagal Login: Cek <i>username</i> dan <i>password</i> anda!”.
Status	Valid

d. Kasus uji *username* salah

Tabel 6.14 Kasus Uji *Username* Salah

Kode Kasus Uji	PV_006
Nama Kasus Uji	Kasus uji <i>username</i> salah
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika <i>username</i> yang dimasukkan salah.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Membuka formulir <i>login</i>. 2. Memasukkan <i>username</i> dengan nilai salah dan <i>password</i> dengan nilai yang benar. 3. Menekan tombol masuk.
Hasil yang diharapkan	Sistem akan menampilkan pesan “Gagal Login: Cek <i>username</i> dan <i>password</i> anda!”.
Hasil Pengujian	Sistem menampilkan pesan “Gagal Login: Cek <i>username</i> dan <i>password</i> anda!”.
Status	Valid

e. Kasus uji *password* salah

Tabel 6.15 Kasus Uji *Password* Salah

Kode Kasus Uji	PV_007
Nama Kasus Uji	Kasus uji <i>password</i> salah
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika pengguna memasukkan <i>password</i> yang salah.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Membuka halaman <i>login</i>. 2. Memasukkan <i>username</i> dengan nilai benar dan <i>password</i> dengan nilai salah. 3. Menekan tombol masuk.

Hasil yang diharapkan	Sistem akan menampilkan pesan “Gagal Login: Cek <i>username</i> dan <i>password</i> anda!”.
Hasil Pengujian	Sistem menampilkan pesan “Gagal Login: Cek <i>username</i> dan <i>password</i> anda!”.
Status	Valid

6.3.3 Pengujian Validasi *Login Backoffice*

- a. Kasus uji *login* sebagai staff *back office*

Tabel 6.16 Kasus Uji *Login* Sebagai Staff *Back office*

Kode Kasus Uji	PV_008
Nama Kasus Uji	Kasus uji <i>login</i> sebagai staff <i>back office</i>
Tujuan Pengujian	Untuk memastikan bahwa staff <i>back office</i> dapat melakukan login ke sistem dengan <i>username</i> , <i>password</i> dan levelnya.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Membuka formulir <i>login</i> staff. 2. Memasukkan <i>username</i> dan <i>password</i> = staffback dan level = Staff. 3. Menekan tombol masuk.
Hasil yang diharapkan	Sistem akan menampilkan halaman daftar pemohon SIUP.
Hasil Pengujian	Sistem menampilkan halaman daftar pemohon SIUP.
Status	Valid

- b. Kasus uji *login* sebagai kepala pengendalian

Tabel 6.17 Kasus Uji *Login* Sebagai Kepala Pengendalian

Kode Kasus Uji	PV_009
Nama Kasus Uji	Kasus uji <i>login</i> sebagai kepala pengendalian
Tujuan Pengujian	Untuk memastikan bahwa kepala pengendalian dapat melakukan <i>login</i> ke sistem dengan <i>username</i> , <i>password</i> dan level.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Membuka formulir pengujian staff. 2. Memasukkan <i>username</i> dan <i>password</i> == pengendalian dan level = Kepala Pengendalian.

	3. Menekan tombol masuk.
Hasil yang diharapkan	Sistem akan menampilkan halaman daftar pemohon.
Hasil Pengujian	Sistem menampilkan halaman daftar pemohon.
Status	Valid

- c. Kasus uji *login* sebagai kepala bidang

Tabel 6.18 Kasus Uji *Login* Sebagai Kepala Bidang

Kode Kasus Uji	PV_010
Nama Kasus Uji	Kasus uji <i>login</i> sebagai kepala bidang
Tujuan Pengujian	Untuk memastikan bahwa kepala bidang dapat melakukan <i>login</i> sesuai dengan <i>username</i> , <i>password</i> dan levelnya
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Membuka formulir <i>login</i>. 2. Memasukkan <i>username</i> dan <i>password</i> = kepalabidang dan level = Kepala Bidang. 3. Menekan tombol masuk.
Hasil yang diharapkan	Sistem akan menampilkan halaman daftar pemohon.
Hasil Pengujian	Sistem menampilkan halaman daftar pemohon.
Status	Valid

- d. Kasus uji *login* sebagai kepala admin

Tabel 6.19 Kasus Uji *Login* Sebagai Kepala Admin

Kode Kasus Uji	PV_011
Nama Kasus Uji	Kasus uji <i>login</i> sebagai kepala admin
Tujuan Pengujian	Untuk memastikan bahwa kepala administrasi dapat melakukan <i>login</i> sesuai dengan <i>username</i> , <i>password</i> dan levelnya.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Membuka formulir <i>login</i>. 2. Memasukkan <i>username</i> dan <i>password</i> = kepalaadmin dan level = Kepala Administrasi. 3. Menekan tombol masuk.

Hasil yang diharapkan	Sistem akan menampilkan halaman daftar pemohon SIUP.
Hasil Pengujian	Sistem menampilkan halaman daftar pemohon SIUP.
Status	Valid

- e. Kasus uji *login* sebagai kepala dinas

Tabel 6.20 Kasus Uji *Login* Sebagai Kepala Dinas

Kode Kasus Uji	PV_012
Nama Kasus Uji	Kasus uji <i>login</i> sebagai kepala dinas
Tujuan Pengujian	Untuk memastikan bahwa kepala dinas dapat melakukan <i>login</i> sesuai dengan <i>username</i> , <i>password</i> dan levelnya.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Membuka formulir <i>login</i>. 2. Memasukkan <i>username</i> dan <i>password</i> = kepaladinas dan level = Kepala Dinas. 3. Menekan tombol masuk.
Hasil yang diharapkan	Sistem akan menampilkan halaman daftar pemohon SIUP.
Hasil Pengujian	Sistem menampilkan halaman daftar pemohon SIUP.
Status	Valid

- f. Kasus uji *login backoffice username* dan *password* kosong

Tabel 6.21 Kasus Uji *Username* dan *Password* Kosong

Kode Kasus Uji	PV_013
Nama Kasus Uji	Kasus uji <i>login backoffice username</i> dan <i>password</i> kosong
Tujuan Pengujian	Untuk memastikan bahwa pengguna tidak dapat melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang kosong serta sistem menampilkan pesan peringatan.
Prosedur Pengujian	<ol style="list-style-type: none"> 3. Membuka formulir <i>login</i>. 4. Menekan tombol masuk.

Hasil yang diharapkan	Sistem akan menampilkan pesan “kolom <i>username</i> wajib diisi, kolom <i>password</i> wajib diisi”.
Hasil Pengujian	Sistem menampilkan pesan “kolom <i>username</i> wajib diisi, kolom <i>password</i> wajib diisi”.
Status	Valid

- g. Kasus uji *login backoffice username* dan *password* tidak sesuai

Tabel 6.22 Kasus Uji *Username* dan *Password* Tidak Sesuai

Kode Kasus Uji	PV_014
Nama Kasus Uji	Kasus uji <i>login backoffice username</i> dan <i>password</i> tidak sesuai
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika pengguna memasukkan data <i>username</i> dan <i>password</i> yang tidak sesuai.
Prosedur Pengujian	4. Membuka formulir <i>login</i> . 5. Memasukkan <i>username</i> dan <i>password</i> yang salah. 6. Menekan tombol login.
Hasil yang diharapkan	Sistem akan menampilkan pesan “Gagal Login: Cek <i>username</i> , <i>password</i> dan level anda!”.
Hasil Pengujian	Sistem menampilkan pesan “Gagal Login: Cek <i>username</i> , <i>password</i> dan level anda!”.
Status	Valid

- h. Kasus uji *login backoffice username* salah

Tabel 6.23 Kasus Uji *Username* Salah

Kode Kasus Uji	PV_015
Nama Kasus Uji	Kasus uji <i>login backoffice username</i> salah
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika <i>username</i> yang dimasukkan salah.
Prosedur Pengujian	4. Membuka formulir <i>login</i> .

	<p>5. Memasukkan <i>username</i> dengan nilai salah dan <i>password</i> dengan nilai yang benar.</p> <p>6. Menekan tombol masuk.</p>
Hasil yang diharapkan	Sistem akan menampilkan pesan “Gagal Login: Cek <i>username</i> , <i>password</i> dan level anda!”.
Hasil Pengujian	Sistem menampilkan pesan “Gagal Login: Cek <i>username</i> , <i>password</i> dan level anda!”.
Status	Valid

- i. Kasus uji *login backoffice password* salah

Tabel 6.24 Kasus Uji *Password* Salah

Kode Kasus Uji	PV_016
Nama Kasus Uji	Kasus uji <i>login backoffice password</i> salah
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika pengguna memasukkan <i>password</i> yang salah.
Prosedur Pengujian	<p>4. Membuka halaman <i>login</i>.</p> <p>5. Memasukkan <i>username</i> dengan nilai benar dan <i>password</i> dengan nilai salah.</p> <p>6. Menekan tombol masuk.</p>
Hasil yang diharapkan	Sistem akan menampilkan pesan “Gagal Login: Cek <i>username</i> , <i>password</i> dan level anda!”.
Hasil Pengujian	Sistem menampilkan pesan “Gagal Login: Cek <i>username</i> , <i>password</i> dan level anda!”.
Status	Valid

- j. Kasus uji level salah

Tabel 6.25 Kasus Uji Level Salah

Kode Kasus Uji	PV_017
Nama Kasus Uji	Kasus uji level salah
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika level yang dimasukkan salah.

Prosedur Pengujian	<ol style="list-style-type: none"> 1. Membuka formulir <i>login</i> 2. Memasukkan <i>username</i> dan <i>password</i> dengan nilai benar dan level dengan nilai salah. 3. Menekan tombol masuk.
Hasil yang diharapkan	Sistem akan menampilkan pesan “Gagal Login: Cek <i>username</i> , <i>password</i> dan level anda!”.
Hasil Pengujian	Sistem akan menampilkan pesan “Gagal Login: Cek <i>username</i> , <i>password</i> dan level anda!”.
Status	Valid

6.3.4 Pengujian Validasi *Logout*

Tabel 6.26 Kasus Uji *Logout*

Kode Kasus Uji	PV_018
Nama Kasus Uji	Kasus Uji <i>logout</i>
Tujuan Pengujian	Untuk memastikan bahwa pengguna dapat keluar dari sistem.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Menekan tombol <i>logout</i>
Hasil yang diharapkan	Sistem akan menghapus <i>session login</i> pemohon dan menampilkan halaman awal pemohon.
Hasil Pengujian	Sistem menghapus <i>session login</i> pemohon dan menampilkan halaman awal pemohon
Status	Valid

6.3.5 Pengujian Validasi Mengupdate Biodata

- a. Kasus uji update biodata tanpa data kosong

Tabel 6.27 Kasus Uji Update Biodata Tanpa Data Kosong

Kode Kasus Uji	PV_019
Nama Kasus Uji	Kasus uji update biodata tanpa data kosong
Tujuan Pengujian	Untuk memastikan bahwa pemohon dapat melakukan edit biodata.

Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu data pemohon 2. Memperbarui nama pemohon, jenis kelamin, tempat lahir, tanggal lahir, alamat, no telp, pekerjaan, kewarganegaraan dan foto 3. Menekan tombol update
Hasil yang diharapkan	Sistem akan menampilkan biodata yang telah diperbarui oleh pemohon.
Hasil Pengujian	Sistem menampilkan biodata yang telah diperbarui oleh pemohon.
Status	Valid

b. Kasus uji *update* biodata ada data kosong

Tabel 6.28 Kasus Uji Edit Biodata Ada Data Kosong

Kode Kasus Uji	PV_020
Nama Kasus Uji	Kasus uji <i>update</i> biodata ada data kosong
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika terdapat form yang tidak diisi oleh pemohon.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu data pemohon 2. Memperbarui nama pemohon saja membiarkan form yang lain kosong. Menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan pada setiap kolom yaitu "kolom wajib diisi!".
Hasil Pengujian	Sistem menampilkan pesan pada setiap kolom yaitu "kolom wajib diisi!".
Status	Valid

a. Kasus uji *update* biodata tanpa upload foto

Tabel 6.29 Kasus Uji Update Biodata Tanpa Upload Foto

Kode Kasus Uji	PV_021
Nama Kasus Uji	Kasus uji <i>update</i> biodata tanpa upload foto

Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika pemohon tidak mengupload foto.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu data pemohon 2. Memperbarui nama pemohon, jenis kelamin, tempat lahir, tanggal lahir, alamat, no telp, pekerjaan, kewarganegaraan dan tanpa upload foto 3. Menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan "Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!".
Hasil Pengujian	Sistem menampilkan pesan "Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!".
Status	Valid

6.3.6 Pengujian Validasi Melihat Notifikasi

- a. Kasus uji ada notifikasi

Tabel 6.30 Kasus Uji Ada Notifikasi

Kode Kasus Uji	PV_022
Nama Kasus Uji	Kasus uji ada notifikasi
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pemberitahuan yang masuk.
Prosedur Pengujian	Menekan tombol pemberitahuan
Hasil yang diharapkan	Sistem akan menampilkan pemberitahuan yang masuk.
Hasil Pengujian	Sistem menampilkan pemberitahuan yang masuk.
Status	Valid

- b. Kasus uji tidak ada notifikasi

Tabel 6.31 Kasus Uji Tidak Ada Notifikasi

Kode Kasus Uji	PV_023
Nama Kasus Uji	Kasus uji tidak ada notifikasi

Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan apabila tidak ada pemberitahuan.
Prosedur Pengujian	Menekan tombol pemberitahuan
Hasil yang diharapkan	Sistem akan menampilkan pesan tidak ada pemberitahuan.
Hasil Pengujian	Sistem menampilkan pesan tidak ada pemberitahuan.
Status	Valid

6.3.7 Pengujian Validasi Mengajukan Perizinan SIUP

- a. Kasus uji data SIUP lengkap

Tabel 6.32 Kasus Uji Data SIUP Lengkap

Kode Kasus Uji	PV_024
Nama Kasus Uji	Kasus uji data SIUP lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat mengupload form pengajuan izin SIUP dengan data lengkap.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu Izin SIUP. 2. Memasukkan identitas perusahaan, kekayaan, kelembagaan, kegiatan usaha, barang dan jasa dagangan utama, akta pendirian, akta perubahan dan mengupload file berkas persyaratan pengajuan izin. 3. Menceklist persetujuan. 4. Menekan tombol submit.
Hasil yang diharapkan	Sistem akan menyimpan data pemohon pengajuan izin siup, menampilkan “pesan permohonan izin berhasil diproses” dan menampilkan halaman status izin.
Hasil Pengujian	Sistem menyimpan data pemohon pengajuan izin siup, menampilkan “pesan permohonan izin berhasil diproses” dan menampilkan halaman status izin.
Status	Valid

- b. Kasus uji pengajuan izin SIUP dengan data tidak lengkap

Tabel 6.33 Kasus Uji Pengajuan Izin SIUP dengan Data tidak Lengkap

Kode Kasus Uji	PV_025
Nama Kasus Uji	Kasus uji pengajuan izin SIUP dengan data tidak lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika form siup yang diupload masih terdapat kolom kosong.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin SIUP. 2. Mengisi semua form kosong kecuali akta pengesahan. 3. Menceklist persetujuan. 4. Menekan tombol submit.
Hasil yang Diharapkan	Sistem akan menampilkan pesan "kolom akta pengesahan wajib diisi!".
Hasil Pengujian	Sistem menampilkan pesan "kolom harus diisi!" pada kolom yang kosong.
Status	Valid

- c. Kasus uji pengajuan izin SIUP tanpa upload berkas

Tabel 6.34 Kasus Uji Pengajuan Izin SIUP Tanpa Upload Berkas

Kode Kasus Uji	PV_026
Nama Kasus Uji	Kasus uji pengajuan izin SIUP tanpa upload berkas
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan apabila form upload berkas masih kosong.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin SIUP. 2. Mengisi formulir data izin seperti identitas perusahaan, kekayaan, kelembagaan, kegiatan usaha, barang dan jasa dagangan utama, akta pendirian dan akta perubahan. 3. Mengupload berkas persyaratan perizinan SIUP. 4. Menceklist persetujuan.

	5. Menekan tombol submit.
Hasil yang Diharapkan	Sistem akan menampilkan halaman formulir izin SIUP dengan pesan kesalahan “Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!”.
Hasil Pengujian	Sistem menampilkan halaman formulir izin SIUP dengan pesan kesalahan “Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!”.
Status	Valid

6.3.8 Pengujian Validasi Mengajukan Perizinan TDP

- a. Kasus uji pengajuan izin TDP dengan data lengkap

Tabel 6.35 Kasus Uji Pengajuan Izin TDP dengan Data Lengkap

Kode Kasus Uji	PV_027
Nama Kasus Uji	Kasus uji pengajuan izin TDP dengan data lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat melakukan upload form perizinan TDP jika dengan data lengkap.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin TDP. 2. Menginputkan semua formulir kosong yang tersedia dan mengupload berkas persyaratan pada setiap kolom upload. 3. Menceklist persetujuan. 4. Menekan tombol submit.
Hasil yang diharapkan	Sistem akan menampilkan pesan “Permohonan izin berhasil di proses” dan menampilkan halaman status izin.
Hasil Pengujian	Sistem menampilkan pesan “Permohonan izin berhasil di proses” dan menampilkan halaman status izin.
Status	Valid

- b. Kasus uji pengajuan izin TDP dengan data tidak lengkap

Tabel 6.36 Kasus Uji Pengajuan Izin TDP dengan Data tidak Lengkap

Kode Kasus Uji	PV_028
Nama Kasus Uji	Kasus uji pengajuan izin TDP dengan data tidak lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika form TDP yang diupload masih terdapat kolom kosong.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin TDP. 2. Menginputkan data pada kolom yang kosong kecuali nama perusahaan. 3. Mengupload berkas persyaratan perizinan TDP. 4. Menceklist persetujuan. 5. Menekan tombol submit.
Hasil yang Diharapkan	Sistem akan menampilkan pesan kesalahan berisi "kolom nama perusahaan wajib diisi!.
Hasil Pengujian	Sistem menampilkan pesan kesalahan berisi "kolom nama perusahaan wajib diisi!".
Status	Valid

- c. Kasus uji pengajuan izin TDP tanpa upload berkas

Tabel 6.37 Kasus Uji Pengajuan Izin TDP tanpa Upload Berkas

Kode Kasus Uji	PV_029
Nama Kasus Uji	Kasus uji pengajuan izin TDP tanpa upload berkas
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan peringatan apabila masih ada kolom upload berkas yang kosong.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin TDP. 2. Menginputkan data pada kolom yang kosong. 3. Menceklist persetujuan. 4. Menekan tombol submit.

Hasil yang Diharapkan	Sistem akan menampilkan pesan “Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!” dan mereload halaman formulir TDP.
Hasil Pengujian	Sistem menampilkan pesan “Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!” dan mereload halaman formulir TDP.
Status	Valid

6.3.9 Pengujian Validasi Mengajukan Perizinan TDUP

- a. Kasus uji pengajuan izin TDUP dengan data lengkap

Tabel 6.38 Kasus Uji Pengajuan Izin TDUP dengan Data Lengkap

Kode Kasus Uji	PV_030
Nama Kasus Uji	Kasus uji pengajuan izin TDUP dengan data lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat melakukan upload form perizinan TDUP apabila data telah lengkap.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin TDUP. 2. Memilih sub menu jenis perizinan TDUP contohnya jenis perizinan TDUP perjalanan wisata. 3. Memasukkan data nama, alamat tempat tinggal, bentuk usaha, nama perusahaan, jabatan, alamat kantor dan jenis usaha. 4. Melakukan upload berkas persyaratan perizinan tdup jenis perjalanan wisata. 5. Menceklist persetujuan. 6. Menekan tombol submit.
Hasil yang diharapkan	Sistem akan menampilkan pesan “Permohonan izin berhasil diproses” dan menampilkan halaman status izin.
Hasil Pengujian	Sistem menampilkan pesan “Permohonan izin berhasil diproses” dan menampilkan halaman status izin.
Status	Valid

- b. Kasus uji pengajuan izin TDUP dengan data tidak lengkap

Tabel 6.39 Kasus Uji Pengajuan Izin TDUP dengan Data Tidak Lengkap

Kode Kasus Uji	PV_031
Nama Kasus Uji	Kasus uji pengajuan izin TDUP dengan data tidak lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan peringatan jika form perizinan TDUP yang di upload masih terdapat kolom yang kosong.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin TDUP. 2. Memilih sub menu jenis perizinan TDUP contohnya jenis perizinan TDUP perjalanan wisata. 3. Memasukkan data nama, alamat tempat tinggal, bentuk usaha, nama perusahaan, jabatan dan jenis usaha. Kolom alamat kantor dibiarkan tetap kosong. 4. Mengupload berkas persyaratan perizinan TDUP jenis perjalanan wisata. 5. Menceklist persetujuan. 6. Menekan tombol submit.
Hasil yang diharapkan	Sistem akan menampilkan pesan kesalahan berupa teks “kolom alamat kantor wajib diisi!” dan mereload halaman formulir perizinan TDUP.
Hasil Pengujian	Sistem menampilkan pesan kesalahan berupa teks “kolom alamat kantor wajib diisi!” dan mereload halaman formulir perizinan TDUP.
Status	Valid

- c. Kasus uji pengajuan izin TDUP tanpa upload berkas

Tabel 6.40 Kasus Uji Pengajuan Izin TDUP tanpa Upload Berkas

Kode Kasus Uji	PV_032
Nama Kasus Uji	Kasus uji pengajuan izin TDUP tanpa upload berkas

Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan peringatan apabila form TDUP yang di upload masih terdapat kolom berkas yang kosong
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin TDUP 2. Memilih sub menu jenis perizinan TDUP contohnya jenis perizinan TDUP perjalanan wisata. 3. Memasukkan data nama, alamat tempat tinggal, bentuk usaha, nama perusahaan, jabatan, alamat kantor dan jenis usaha. 4. Menceklist persetujuan. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem akan menampilkan pesan kesalahan berupa "Gagal! Gambar belum di upload, upload gambar terlebih dahulu dan menampilkan halaman formulir izin TDUP".
Hasil Pengujian	Sistem menampilkan pesan kesalahan berupa "Gagal! Gambar belum di upload, upload gambar terlebih dahulu dan menampilkan halaman formulir izin TDUP".
Status	Valid

6.3.10 Pengujian Validasi Mengajukan Perizinan Reklame

- a. Kasus uji pengajuan izin reklame dengan data lengkap

Tabel 6.41 Kasus Uji Pengajuan Izin Reklame dengan Data Lengkap

Kode Kasus Uji	PV_033
Nama Kasus Uji	Kasus uji pengajuan izin reklame dengan data lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat melakukan upload form reklame apabila data yang dimasukkan telah lengkap.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin reklame. 2. Memilih jenis izin reklame seperti reklame insidental. 3. Memasukkan data pada formulir reklame insidental yang terdiri dari nama pemohon, alamat pemohon, nama

	<p>perusahaan, jabatan di perusahaan, alamat perusahaan, no telp perusahaan, bentuk reklame, isi reklame, ukuran reklame, jumlah, ketinggian, lama pemasangan mulai dan sampai dan lokasi pemasangan.</p> <p>4. Melakukan upload berkas persyaratan perizinan reklame jenis reklame insidental.</p> <p>5. Menceklist persetujuan.</p> <p>6. Menekan tombol submit.</p>
Hasil yang diharapkan	Sistem akan menampilkan pesan "Permohonan izin berhasil di proses" dan menampilkan halaman status izin.
Hasil Pengujian	Sistem menampilkan pesan "Permohonan izin berhasil di proses" dan menampilkan halaman status izin.
Status	Valid

- b. Kasus uji pengajuan izin reklame dengan data tidak lengkap

Tabel 6.42 Kasus Uji Pengajuan Izin Reklame dengan Data tidak Lengkap

Kode Kasus Uji	PV_034
Nama Kasus Uji	Kasus uji pengajuan izin reklame dengan data tidak lengkap
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika form reklame yang di upload masih terdapat kolom yang kosong.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu izin reklame. 2. Memilih jenis izin reklame seperti reklame insidental. 3. Memasukkan data pada formulir reklame insidental yang terdiri dari nama pemohon, alamat pemohon, nama perusahaan, jabatan di perusahaan, alamat perusahaan, no telp perusahaan, bentuk reklame, isi reklame, ukuran reklame, jumlah, ketinggian, lama pemasangan mulai dan sampai dan

	<p>membiarkan kolom lokasi pemasangan tetap kosong.</p> <p>4. Melakukan upload berkas persyaratan perizinan reklame jenis reklame insidental.</p> <p>5. Menceklist persetujuan.</p> <p>6. Menekan tombol submit.</p>
Hasil yang diharapkan	Sistem akan mereload halaman formulir reklame insidental dan menampilkan pesan kesalahan “kolom lokasi pemasangan wajib diisi!”
Hasil Pengujian	Sistem mereload halaman formulir reklame insidental dan menampilkan pesan kesalahan “kolom lokasi pemasangan wajib diisi!”
Status	Valid

c. Kasus uji pengajuan izin reklame tanpa upload berkas

Tabel 6.43 Kasus Uji Pengajuan Izin Reklame Tanpa Upload Berkas

Kode Kasus Uji	PV_035
Nama Kasus Uji	Kasus uji pengajuan izin reklame tanpa upload berkas
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan pesan kesalahan jika form reklame yang di upload terdapat kolom berkas yang kosong.
Prosedur Pengujian	<ol style="list-style-type: none"> Memilih menu izin reklame. Memilih jenis izin reklame seperti reklame insidental. Memasukkan data pada formulir reklame insidental yang terdiri dari nama pemohon, alamat pemohon, nama perusahaan, jabatan di perusahaan, alamat perusahaan, no telp perusahaan, bentuk reklame, isi reklame, ukuran reklame, jumlah, ketinggian, lama pemasangan mulai dan sampai dan lokasi pemasangan. Menceklist persetujuan. Menekan tombol submit.

Hasil yang diharapkan	Sistem akan mereload halaman formulir reklame insidental dan menampilkan pesan kesalahan berupa “Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!”.
Hasil Pengujian	Sistem mereload halaman formulir reklame insidental dan menampilkan pesan kesalahan berupa “Gagal! Gambar Belum di Upload, Upload Gambar Terlebih Dahulu!”.
Status	Valid

6.3.11 Pengujian Validasi Melihat Status Perizinan

Tabel 6.44 Kasus Uji Melihat Status Perizinan

Kode Kasus Uji	PV_036
Nama Kasus Uji	Kasus uji melihat status perizinan
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan status perizinan pemohon yang telah melakukan pengajuan izin.
Prosedur Pengujian	1. Memilih menu permohonan pengajuan. 2. Menekan tombol jenis izin yang diinginkan
Hasil yang diharapkan	Sistem akan menampilkan halaman status permohonan izin yang diajukan oleh pemohon.
Hasil Pengujian	Sistem menampilkan halaman status permohonan izin yang diajukan oleh pemohon.
Status	Valid

6.3.12 Pengujian Validasi Memeriksa Kelengkapan Berkas

- a. Kasus uji memeriksa kelengkapan berkas SIUP

Tabel 6.45 Kasus Uji Memeriksa Kelengkapan Berkas SIUP

Kode Kasus Uji	PV_037
Nama Kasus Uji	Kasus uji memeriksa kelengkapan berkas SIUP
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menampilkan berkas pemohon dan dapat melakukan pemeriksaan berkas dengan menceklist pada kolom yang disediakan.

Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon SIUP. 2. Menekan tombol details. 3. Mencentang pada kolom sesuai kebenaran berkas yang dilampirkan pemohon. 4. Mengisi catatan, jika memang ada catatan untuk pemohon. 5. Menekan tombol validasi.
Hasil yang diharapkan	Sistem akan menampilkan halaman draft untuk mengisi form pembuatan SK perizinan SIUP.
Hasil Pengujian	Sistem menampilkan halaman draft untuk mengisi form pembuatan SK perizinan SIUP.
Status	Valid

- b. Kasus uji memeriksa kelengkapan berkas TDP

Tabel 6.46 Kasus Uji Memeriksa Kelengkapan Berkas TDP

Kode Kasus Uji	PV_038
Nama Kasus Uji	Kasus uji memeriksa kelengkapan berkas TDP
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menampilkan berkas pemohon dan dapat melakukan pemeriksaan berkas dengan menceklist pada kolom yang disediakan.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon TDP. 2. Menekan tombol details. 3. Mencentang pada kolom sesuai kebenaran berkas yang dilampirkan pemohon. 4. Mengisi catatan, jika memang ada catatan untuk pemohon. 5. Menekan tombol validasi.
Hasil yang diharapkan	Sistem akan menampilkan halaman draft untuk mengisi form pembuatan SK perizinan TDP.
Hasil Pengujian	Sistem menampilkan halaman draft untuk mengisi form pembuatan SK perizinan TDP.
Status	Valid

c. Kasus uji memeriksa kelengkapan berkas TDUP

Tabel 6.47 Kasus Uji Memeriksa Kelengkapan Berkas TDUP

Kode Kasus Uji	PV_039
Nama Kasus Uji	Kasus uji memeriksa kelengkapan berkas TDUP
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menampilkan berkas pemohon dan dapat melakukan pemeriksaan berkas dengan menceklist pada kolom yang disediakan.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon TDUP. 2. Memilih jenis TDUP yang akan diperiksa kelengkapannya. 3. Menekan tombol details. 4. Mencentang pada kolom sesuai kebenaran berkas yang dilampirkan pemohon. 5. Mengisi catatan, jika memang ada catatan untuk pemohon. 6. Menekan tombol validasi.
Hasil yang diharapkan	Sistem akan menampilkan halaman draft untuk mengisi form pembuatan SK perizinan TDUP.
Hasil Pengujian	Sistem menampilkan halaman draft untuk mengisi form pembuatan SK perizinan TDUP.
Status	Valid

d. Kasus uji memeriksa kelengkapan berkas reklame

Tabel 6.48 Kasus Uji Memeriksa Kelengkapan Berkas Reklame

Kode Kasus Uji	PV_040
Nama Kasus Uji	Kasus uji memeriksa kelengkapan berkas reklame
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menampilkan berkas pemohon dan dapat melakukan pemeriksaan berkas dengan menceklist pada kolom yang disediakan.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon reklame

	<ol style="list-style-type: none"> Memilih jenis reklame yang akan diperiksa kelengkapannya. Menekan tombol details. Mencentang pada kolom sesuai kebenaran berkas yang dilampirkan pemohon. Mengisi catatan, jika memang ada catatan untuk pemohon. Menekan tombol validasi.
Hasil yang diharapkan	Sistem akan menampilkan halaman draft untuk mengisi form pembuatan SK perizinan reklame.
Hasil Pengujian	Sistem menampilkan halaman draft untuk mengisi form pembuatan SK perizinan reklame.
Status	Valid

6.3.13 Pengujian Validasi Membuat SK Perizinan

- a. Kasus uji membuat SK perizinan SIUP

Tabel 6.49 Kasus Uji membuat SK perizinan SIUP

Kode Kasus Uji	PV_041
Nama Kasus Uji	Kasus uji membuat SK perizinan SIUP
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan form SK perizinan SIUP yang telah berisi data pemohon.
Prosedur Pengujian	<ol style="list-style-type: none"> Telah melakukan validasi pemeriksaan berkas pemohon SIUP. Mengisi form draft SK perizinan SIUP. Menekan tombol submit.
Hasil yang Diharapkan	Sistem akan menyimpan data SK perizinan ke dalam database dan menampilkan halaman daftar pemohon SIUP.
Hasil Pengujian	Sistem menyimpan data SK perizinan ke dalam database dan menampilkan halaman daftar pemohon SIUP.
Status	Valid

b. Kasus uji membuat SK perizinan TDP

Tabel 6.50 Kasus Uji membuat SK perizinan TDP

Kode Kasus Uji	PV_042
Nama Kasus Uji	Kasus uji membuat SK perizinan TDP
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan form SK perizinan TDP yang telah berisi data pemohon.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Telah melakukan validasi pemeriksaan berkas pemohon TDP. 2. Mengisi form draft SK perizinan TDP. 3. Menekan tombol submit.
Hasil yang Diharapkan	Sistem akan menyimpan data SK perizinan TDP ke dalam database dan menampilkan daftar pemohon TDP.
Hasil Pengujian	Sistem menyimpan data SK perizinan TDP ke dalam database dan menampilkan daftar pemohon TDP.
Status	Valid

c. Kasus uji membuat SK perizinan TDUP

Tabel 6.51 Kasus Uji membuat SK perizinan TDUP

Kode Kasus Uji	PV_043
Nama Kasus Uji	Kasus uji membuat SK perizinan TDUP
Tujuan Pengujian	Untuk memastikan bahwa sistem akan menampilkan form SK perizinan TDUP yang telah berisi data pemohon.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Telah melakukan validasi pemeriksaan berkas pemohon. 2. Mengisi form draft SK perizinan TDUP. 3. Menekan tombol submit.
Hasil yang Diharapkan	Sistem akan menyimpan data SK perizinan TDUP ke dalam database dan menampilkan halaman daftar pemohon.

Hasil Pengujian	Sistem menyimpan data SK perizinan TDUP ke dalam database dan menampilkan halaman daftar pemohon.
Status	Valid

- d. Kasus uji membuat SK perizinan Reklame

Tabel 6.52 Kasus Uji membuat SK perizinan Reklame

Kode Kasus Uji	PV_044
Nama Kasus Uji	Kasus uji membuat SK perizinan Reklame
Tujuan Perizinan	Untuk memastikan bahwa sistem akan menampilkan form SK perizinan reklame yang telah berisi data pemohon.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Telah melakukan validasi pemeriksaan berkas pemohon. 2. Mengisi form draft SK perizinan reklame. 3. Menekan tombol submit.
Hasil yang Diharapkan	Sistem akan menyimpan data SK perizinan reklame ke dalam database dan menampilkan halaman daftar pemohon reklame.
Hasil Pengujian	Sistem menyimpan data SK perizinan reklame ke dalam database dan menampilkan halaman daftar pemohon reklame.
Status	Valid

6.3.14 Pengujian Validasi Mencetak SK Perizinan

- a. Kasus uji mencetak SK perizinan SIUP

Tabel 6.53 Kasus Uji Mencetak SK Perizinan SIUP

Kode Kasus Uji	PV_045
Nama Kasus Uji	Kasus uji mencetak SK perizinan SIUP
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menampilkan halaman SK perizinan SIUP yang akan dicetak.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu cetak SK. 2. Memilih menu SIUP. 3. Menekan tombol cetak.

Hasil yang Diharapkan	Sistem akan menampilkan SK perizinan SIUP pemohon yang siap dicetak.
Hasil Pengujian	Sistem menampilkan SK perizinan SIUP pemohon yang siap dicetak.
Status	Valid

- b. Kasus uji mencetak SK perizinan tdp

Tabel 6.54 Kasus Uji Mencetak SK Perizinan TDP

Kode Kasus Uji	PV_046
Nama Kasus Uji	Kasus uji mencetak SK perizinan TDP
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menampilkan halaman SK perizinan TDP yang akan dicetak.
Prosedur Pengujian	1. Memilih menu cetak SK. 2. Memilih menu TDP. 3. Menekan tombol cetak.
Hasil yang Diharapkan	Sistem akan menampilkan SK perizinan TDP pemohon yang siap dicetak.
Hasil Pengujian	Sistem menampilkan SK perizinan TDP pemohon yang siap dicetak.
Status	Valid

- c. Kasus uji mencetak SK perizinan TDUP

Tabel 6.55 Kasus Uji Mencetak SK Perizinan TDUP

Kode Kasus Uji	PV_047
Nama Kasus Uji	Kasus uji mencetak SK perizinan TDUP
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menampilkan halaman SK perizinan TDUP yang akan dicetak.
Prosedur Pengujian	1. Memilih menu cetak SK. 2. Memilih menu TDUP. 3. Menekan tombol cetak.
Hasil yang Diharapkan	Sistem akan menampilkan SK perizinan TDUP pemohon yang siap dicetak.

Hasil Pengujian	Sistem menampilkan SK perizinan TDUP pemohon yang siap dicetak.
Status	Valid

- d. Kasus uji mencetak SK perizinan reklame

Tabel 6.56 Kasus Uji Mencetak SK Perizinan Reklame

Kode Kasus Uji	PV_048
Nama Kasus Uji	Kasus uji mencetak SK perizinan reklame
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menampilkan halaman SK perizinan reklame yang akan dicetak.
Prosedur Pengujian	1. Memilih menu cetak SK. 2. Memilih menu reklame. 3. Menekan tombol cetak.
Hasil yang Diharapkan	Sistem akan menampilkan SK perizinan reklame pemohon yang siap dicetak.
Hasil Pengujian	Sistem menampilkan SK perizinan reklame pemohon yang siap dicetak.
Status	Valid

6.3.15 Pengujian Validasi Memberikan Nomor Surat

- a. Kasus uji memberikan nomor surat SIUP

Tabel 6.57 Kasus Uji Memberikan Nomor Surat SIUP

Kode Kasus Uji	PV_049
Nama Kasus Uji	Kasus uji memberikan nomor surat SIUP
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menampilkan form SK perizinan SIUP untuk kepala admin dapat memasukkan nomor surat.
Prosedur Pengujian	1. Memilih menu daftar pemohon izin SIUP. 2. Menekan tombol edit data. 3. Mengisi nomor surat pada kolom nomor. 4. Menekan tombol validasi.

Hasil yang Diharapkan	Sistem akan menampilkan pesan bahwa data berhasil di <i>update</i> .
Hasil Pengujian	Sistem menampilkan pesan bahwa data berhasil di <i>update</i> .
Status	Valid

- b. Kasus uji memberikan nomor surat TDP

Tabel 6.58 Kasus Uji Memberikan Nomor Surat TDP

Kode Kasus Uji	PV_050
Nama Kasus Uji	Kasus uji memberikan nomor surat TDP
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menampilkan form SK perizinan TDP untuk kepala admin dapat memasukkan nomor surat.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon izin TDP. 2. Menekan tombol edit data. 3. Mengisi nomor surat pada kolom nomor. 4. Menekan tombol update.
Hasil yang Diharapkan	Sistem akan menampilkan pesan bahwa data berhasil di update.
Hasil Pengujian	Sistem menampilkan pesan bahwa data berhasil di update.
Status	Valid

- c. Kasus uji memberikan nomor surat tdup

Tabel 6.59 Kasus Uji Memberikan Nomor Surat TDUP

Kode Kasus Uji	PV_051
Nama Kasus Uji	Kasus uji memberikan nomor surat tdup
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menampilkan form SK perizinan TDUP untuk kepala admin dapat memasukkan nomor surat.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon TDUP. 2. Memilih sub jenis TDUP (TDUP wisata atau TDUP makanan).

	3. Menekan tombol edit data. 4. Mengisi nomor surat pada kolom nomor. 5. Menekan tombol update.
Hasil yang Diharapkan	Sistem akan menampilkan pesan bahwa data berhasil di update.
Hasil Pengujian	Sistem menampilkan pesan bahwa data berhasil di update.
Status	Valid

- d. Kasus uji memberikan nomor surat reklame

Tabel 6.60 Kasus Uji Memberikan Nomor Surat Reklame

Kode Kasus Uji	PV_052
Nama Kasus Uji	Kasus uji memberikan nomor surat reklame
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menampilkan form SK perizinan reklame untuk kepala admin dapat memasukkan nomor surat.
Prosedur Pengujian	1. Memilih menu daftar pemohon reklame. 2. Menekan tombol edit data. 3. Mengisi nomor surat pada kolom nomor. 4. Menekan tombol update.
Hasil yang Diharapkan	Sistem akan menampilkan pesan data berhasil di <i>update</i> .
Hasil Pengujian	Sistem menampilkan pesan data berhasil di <i>update</i> .
Status	Valid

6.3.16 Pengujian Validasi Menvalidasi Berkas Perizinan

- a. Kasus uji menvalidasi berkas perizinan SIUP

Tabel 6.61 Kasus Uji Menvalidasi Berkas Perizinan SIUP

Kode Kasus Uji	PV_053
Nama Kasus Uji	Kasus uji menvalidasi berkas perizinan SIUP

Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat melakukan validasi pada berkas perizinan SIUP.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon SIUP. 2. Menekan tombol details. 3. Mencentang kesesuaian pada identitas pemohon, identitas perusahaan, legalitas perusahaan dan berkas yang dilampirkan. 4. Memasukkan data catatan bila diperlukan. 5. Menekan tombol validasi.
Hasil yang Diharapkan	Sistem akan menampilkan halaman daftar pemohon SIUP. Data catatan akan tersimpan di database sesuai dengan id pemohon.
Hasil Pengujian	Sistem menampilkan halaman daftar pemohon SIUP. Data catatan akan tersimpan di database sesuai dengan id pemohon.
Status	Valid

b. Kasus uji menvalidasi berkas perizinan TDP

Tabel 6.62 Kasus Uji Menvalidasi Berkas Perizinan TDP

Kode Kasus Uji	PV_054
Nama Kasus Uji	Kasus uji menvalidasi berkas perizinan TDP
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat melakukan validasi pada berkas perizinan TDP.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon TDP. 2. Menekan tombol details. 3. Mencentang kesesuaian pada identitas pemohon, identitas perusahaan, jenis perizinan yang dimiliki, legalitas perusahaan, berkas yang dilampirkan. 4. Memasukkan data catatan bila diperlukan. 5. Menekan tombol validasi.
Hasil yang Diharapkan	Sistem akan menampilkan halaman daftar pemohon TDP. Data catatan akan tersimpan di <i>database</i> sesuai dengan id pemohon.

Hasil Pengujian	Sistem menampilkan halaman daftar pemohon TDP. Data catatan akan tersimpan di <i>database</i> sesuai dengan id pemohon.
Status	Valid

- c. Kasus uji menvalidasi berkas perizinan tdup

Tabel 6.63 Kasus Uji Menvalidasi Berkas Perizinan TDUP

Kode Kasus Uji	PV_055
Nama Kasus Uji	Kasus uji menvalidasi berkas perizinan tdup
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat melakukan validasi pada berkas perizinan TDUP.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon TDUP. 2. Menekan tombol details. 3. Mencentang kesesuaian pada identitas pemohon, identitas usaha dan berkas yang dilampirkan. 4. Memasukkan data catatan. 5. Menekan tombol validasi.
Hasil yang Diharapkan	Sistem akan menampilkan halaman daftar pemohon TDP. Data catatan akan tersimpan di <i>database</i> sesuai dengan id pemohon.
Hasil Pengujian	Sistem akan menampilkan halaman daftar pemohon TDP. Data catatan akan tersimpan di <i>database</i> sesuai dengan id pemohon.
Status	Valid

- d. Kasus uji menvalidasi berkas perizinan reklame

Tabel 6.64 Kasus Uji Menvalidasi Berkas Perizinan Reklame

Kode Kasus Uji	PV_056
Nama Kasus Uji	Kasus uji menvalidasi berkas perizinan reklame
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat melakukan validasi pada berkas perizinan reklame.

Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon reklame. 2. Menekan tombol details. 3. Mencentang kesesuaian pada identitas pemohon, identitas pemasangan reklame dan berkas yang dilampirkan. 4. Memasukkan data catatan. 5. Menekan tombol validasi.
Hasil yang Diharapkan	Sistem akan menampilkan halaman daftar pemohon reklame. Data catatan akan tersimpan di <i>database</i> sesuai dengan id pemohon.
Hasil Pengujian	Sistem menampilkan halaman daftar pemohon reklame. Data catatan akan tersimpan di <i>database</i> sesuai dengan id pemohon.
Status	Valid

6.3.17 Pengujian Validasi Menerbitkan SK Perizinan

- a. Kasus uji menerbitkan SK perizinan SIUP

Tabel 6.65 Kasus Uji Menerbitkan SK Perizinan SIUP

Kode Kasus Uji	PV_057
Nama Kasus Uji	Kasus uji menerbitkan SK perizinan SIUP
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menerbitkan SK perizinan SIUP.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon SIUP. 2. Menekan tombol terbitkan. 3. Menekan tombol iya pada pesan konfirmasi.
Hasil yang Diharapkan	Sistem akan memperbarui id status pemohon dengan nilai 6 dan menampilkan halaman daftar pemohon SIUP.
Hasil Pengujian	Sistem memperbarui id status pemohon dengan nilai 6 dan menampilkan halaman daftar pemohon SIUP.
Status	Valid

b. Kasus uji menerbitkan SK perizinan TDP

Tabel 6.66 Kasus Uji Menerbitkan SK Perizinan TDP

Kode Kasus Uji	PV_058
Nama Kasus Uji	Kasus uji menerbitkan SK perizinan TDP
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menerbitkan SK perizinan TDP.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon TDP. 2. Menekan tombol terbitkan. 3. Menekan tombol iya pada pesan konfirmasi.
Hasil yang Diharapkan	Sistem akan memperbarui id status pemohon menjadi nilai 6 dan menampilkan halaman daftar pemohon TDP.
Hasil Pengujian	Sistem memperbarui id status pemohon menjadi nilai 6 dan menampilkan halaman daftar pemohon TDP.
Status	Valid

c. Kasus uji menerbitkan SK perizinan TDUP

Tabel 6.67 Kasus Uji Menerbitkan SK Perizinan TDUP

Kode Kasus Uji	PV_059
Nama Kasus Uji	Kasus uji menerbitkan SK perizinan TDUP
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menerbitkan SK perizinan TDUP.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon TDUP. 2. Memilih sub jenis perizinan TDUP wisata atau TDUP makanan. 3. Menekan tombol terbitkan. 4. Menekan tombol iya pada pesan konfirmasi.
Hasil yang Diharapkan	Sistem akan memperbarui id status pemohon menjadi nilai 6 dan menampilkan halaman daftar pemohon TDUP.

Hasil Pengujian	Sistem akan memperbarui id status pemohon menjadi nilai 6 dan menampilkan halaman daftar pemohon TDUP.
Status	Valid

d. Kasus uji menerbitkan SK perizinan reklame

Tabel 6.68 Kasus Uji Menerbitkan SK Perizinan Reklame

Kode Kasus Uji	PV_060
Nama Kasus Uji	Kasus uji menerbitkan SK perizinan reklame
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menerbitkan SK perizinan reklame.
Prosedur Pengujian	1. Memilih menu daftar pemohon reklame. 2. Menekan tombol terbitkan. 3. Menekan tombol iya pada pesan konfirmasi.
Hasil yang Diharapkan	Sistem akan memperbarui id status pemohon menjadi 6 dan menampilkan halaman daftar pemohon reklame.
Hasil Pengujian	Sistem memperbarui id status pemohon menjadi 6 dan menampilkan halaman daftar pemohon reklame.
Status	Valid

6.3.18 Pengujian Validasi Melihat Daftar Pemohon

Tabel 6.69 Kasus Uji Melihat Daftar Pemohon

Kode Kasus Uji	PV_061
Nama Kasus Uji	Kasus uji melihat daftar pemohon
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menampilkan daftar pemohon yang terdaftar di sistem.
Prosedur Pengujian	1. Memilih menu daftar pemohon
Hasil yang diharapkan	Sistem akan menampilkan halaman daftar pemohon yang terdaftar di dalam sistem.
Hasil Pengujian	Sistem menampilkan halaman daftar pemohon yang terdaftar di dalam sistem.

Status	Valid
--------	-------

6.3.19 Pengujian Validasi Menghapus Akun Pemohon

Tabel 6.70 Kasus Uji Menghapus Akun Pemohon

Kode Kasus Uji	PV_062
Nama Kasus Uji	Kasus uji menghapus akun pemohon
Tujuan Pengujian	Untuk membuktikan bahwa sistem dapat menghapus data pemohon yang terdaftar di sistem.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Memilih menu daftar pemohon. 2. Menekan tombol delete. 3. Menekan tombol ok pada pesan konfirmasi
Hasil yang diharapkan	Sistem akan menampilkan pesan “data pemohon berhasil di hapus”. Data pemohon di database juga ikut terhapus serta sistem akan mereload halaman daftar pemohon.
Hasil Pengujian	Sistem menampilkan pesan “data pemohon berhasil di hapus”. Data pemohon di database juga ikut terhapus serta sistem akan mereload halaman daftar pemohon.
Status	Valid

6.3.20 Pengujian *Compatibility*

Dalam kasus ini pengujian nonfungsional dilakukan menggunakan jenis pengujian *compatibility* yaitu untuk memeriksa apakah sistem perizinan dapat dijalankan pada berbagai perangkat *browser* seperti *google chrome*, Mozilla *firefox* dan *internet explorer*. Tujuan dari pengujian ini adalah untuk mengungkapkan kesalahan atau masalah-masalah eksekusi yang dapat ditelusuri melalui perbedaan konfigurasi dari setiap *browser*. Parameter yang digunakan dalam pengujian ini adalah standarisasi html, tata letak halaman, *support template bootstrap*, *font style* dan *responsive web*. Hasil pengujian *compatibility* ditunjukkan pada Tabel 6.67. Pada Gambar 6.17, Gambar 6.18 dan Gambar 6.19 merupakan antarmuka sistem perizinan dari hasil pengujian *compatibility* yang mengambil salah satu fitur sistem yaitu pengajuan izin SIUP.

Tabel 6.71 Hasil Pengujian *Compatibility*

Kode Kasus Uji	Hasil Pengujian		
	<i>Google Chrome</i>	<i>Mozilla Firefox</i>	<i>Internet Explorer</i>
PV_001	Valid	Valid	Valid
PV_002	Valid	Valid	Valid
PV_003	Valid	Valid	Valid
PV_004	Valid	Valid	Valid
PV_005	Valid	Valid	Valid
PV_006	Valid	Valid	Valid
PV_007	Valid	Valid	Valid
PV_008	Valid	Valid	Valid
PV_009	Valid	Valid	Valid
PV_010	Valid	Valid	Valid
PV_011	Valid	Valid	Valid
PV_012	Valid	Valid	Valid
PV_013	Valid	Valid	Valid
PV_014	Valid	Valid	Valid
PV_015	Valid	Valid	Valid
PV_016	Valid	Valid	Valid
PV_017	Valid	Valid	Valid
PV_018	Valid	Valid	Valid
PV_019	Valid	Valid	Valid
PV_020	Valid	Valid	Valid
PV_021	Valid	Valid	Valid
PV_022	Valid	Valid	Valid
PV_023	Valid	Valid	Valid
PV_024	Valid	Valid	Valid
PV_025	Valid	Valid	Valid
PV_026	Valid	Valid	Valid
PV_027	Valid	Valid	Valid
PV_028	Valid	Valid	Valid

PV_029	Valid	Valid	Valid
PV_030	Valid	Valid	Valid
PV_031	Valid	Valid	Valid
PV_032	Valid	Valid	Valid
PV_033	Valid	Valid	Valid
PV_034	Valid	Valid	Valid
PV_035	Valid	Valid	Valid
PV_036	Valid	Valid	Valid
PV_037	Valid	Valid	Valid
PV_038	Valid	Valid	Valid
PV_039	Valid	Valid	Valid
PV_040	Valid	Valid	Valid
PV_041	Valid	Valid	Valid
PV_042	Valid	Valid	Valid
PV_043	Valid	Valid	Valid
PV_044	Valid	Valid	Valid
PV_045	Valid	Valid	Valid
PV_046	Valid	Valid	Valid
PV_047	Valid	Valid	Valid
PV_048	Valid	Valid	Valid
PV_049	Valid	Valid	Valid
PV_050	Valid	Valid	Valid
PV_051	Valid	Valid	Valid
PV_052	Valid	Valid	Valid
PV_053	Valid	Valid	Valid
PV_054	Valid	Valid	Valid
PV_055	Valid	Valid	Valid
PV_056	Valid	Valid	Valid
PV_057	Valid	Valid	Valid
PV_058	Valid	Valid	Valid
PV_059	Valid	Valid	Valid

PV_060	Valid	Valid	Valid
PV_061	Valid	Valid	Valid
PV_062	Valid	Valid	Valid

Gambar 6.17 Sistem Perizinan Pada Browser Chrome

Gambar 6.18 Sistem Perizinan Pada Browser Mozilla Firefox

http://localhost/perizinan/permohonan_siup/action_siup

Sistem Perizinan

Hi hinduniva

Halaman Awal

Data Pemohon

Permohonan Pengajuan

Izin SIUP

Izin TDP

Izin TDUP

Izin Reklame

Sistem Perizinan / Izin SIUP / Form SIUP

Tambah Data SIUP

Nama Perusahaan

Nama Penanggung jawab

Jabatan

Alamat Perusahaan

No Telepon / Fax

Status

Kekayaan Bersih Perusahaan(Tidak termasuk tanah dan bangunan)

Gambar 6.19 Sistem Perizinan Pada Browser Internet Explorer

Berdasarkan dari hasil pengujian yang ditunjukkan pada Tabel 6.67 tersebut maka sistem perizinan dapat diimplementasikan pada tiga *browser* yaitu *google chrome*, *mozilla firefox* dan *internet explorer* sehingga dinyatakan valid karena telah sesuai dengan hasil yang diharapkan.



BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan dari hasil analisis kebutuhan, perancangan, implementasi dan pengujian yang telah dilakukan pada penelitian ini maka dapat diambil kesimpulan sebagai berikut:

1. Pada bagian analisis kebutuhan untuk sistem perizinan menghasilkan 19 kebutuhan fungsional dari 7 aktor yaitu pengguna, pemohon, staff *back office*, kepala pengendalian, kepala bidang, kepala administrasi dan kepala dinas. Kebutuhan fungsional yang didapatkan yaitu *login*, mengajukan izin SIUP, TDP, TDUP, Reklame, memvalidasi perizinan, melihat status izin serta kebutuhan yang lainnya. Untuk kebutuhan nonfungsional hasil dari analisis menghasilkan 1 kebutuhan yaitu *compatibility*.
2. Pada bagian perancangan yang telah dilakukan didapatkan perancangan arsitektur, pemodelan diagram *sequence*, pemodelan diagram klas dan menghasilkan perancangan data yang dapat dijadikan acuan dalam implementasi basis data melalui perancangan ERD, CDM dan PDM. Pada bagian implementasi dihasilkan sistem perizinan dengan fitur fitur utama seperti mengajukan perizinan SIUP, TDP, TDUP dan Reklame, memvalidasi perizinan, melihat status perizinan dan melihat notifikasi.
3. Pada bagian pengujian dibagi menjadi dua yaitu pengujian kebutuhan fungsional dan kebutuhan nonfungsional. Pada pengujian fungsional dilakukan menggunakan teknik *whitebox* dan *blackbox* didapatkan hasil kebutuhan yang valid dimana sistem telah berjalan sesuai dengan analisis kebutuhan dan perancangan yang didapatkan. Pengujian *whitebox* yang dilakukan yaitu pengujian unit dan integrasi. Pada pengujian unit dan integrasi sistem berhasil melewati semua jalur pengujian. Pada pengujian validasi dilakukan dengan pengujian *blackbox* terhadap 19 kebutuhan fungsional dan dihasilkan 19 kebutuhan fungsional yang valid sehingga pengujian validasi dinyatakan 100% valid. Untuk pengujian kebutuhan nonfungsional dilakukan menggunakan pengujian *compatibility* yang menghasilkan nilai 100% valid untuk *browser mozilla firefox, google chrome, internet explorer*.

7.2 Saran

Saran yang dapat diberikan dari hasil penelitian ini untuk mendapatkan penelitian yang lebih baik kedepannya adalah pada fitur pengajuan izin reklame, pemohon tidak mendapatkan jumlah pajak secara langsung karena sistem belum terintegrasi dengan dinas lain. Alangkah lebih baik jika sistem dapat terintegrasi secara langsung dengan dinas yang mengurus masalah pajak reklame sehingga pemohon dapat mengetahui berapa nilai pajak yang harus dibayarkan.

DAFTAR PUSTAKA

- Akbar, C. K., Hanafi, D. I. & Makmur, D. M., 2012. *Kebijakan Perijinan*. Malang: Universitas Brawijaya Press.
- Axmark, D. & Widenius, M. M., 1997. *Reference Manual for the MySQL Database System version 5.7*. U.S: Oracle.
- Chunaivi, M., 2017. *Pembangunan Sistem Perizinan Pada Dinas Kementrian Komunikasi dan Informatika* [Interview] (18 October 2017).
- dpmpstsp, 2018. dpmpstsp. [Online] Available at: <http://dpmpstsp.pasuruankota.go.id/visi-misi/> [Accessed 15 January 2018].
- Fowler, M., 2003. *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. 3rd ed. Boston: Addison Wesley.
- Inpres RI No 3 Tahun 2003, 2003. *Kebijakan dan Strategi Nasional Pengembangan E-government*, Jakarta: s.n.
- Nugroho, B., 2004. *PHP dan MySQL dengan editor Dreamweaver MX*. 1st ed. Yogyakarta: Penerbit Andi.
- Peraturan Walikota, 2011. *Tugas Pokok dan Fungsi Badan Penanaman Modal dan Pelayanan Perijinan Terpadu*, Pasuruan: s.n.
- Pressman, R. S., 2010. *Software Engineering A Practitioner's Approach*. 7th ed. New York: The McGraw-Hill Companies, Inc.
- Pusat Kajian, M. P., 2011. *Modul Analisis Proses Bisnis Instansi Pemerintah*. Jakarta: Lembaga Administrasi Negara.
- Rosa & Shalahuddin, M., 2015. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Rumbaugh, J., Jacobson, I. & Booch, G., 2004. *The Unified Modeling Language Reference Manual*. 2nd ed. Boston: Pearson Education, Inc.
- Siahaan, D., 2012. *Analisa Kebutuhan Dalam Rekayasa Perangkat Lunak*. Yogyakarta: Penerbit Andi.
- Sommerville, I., 2011. *Software Engineering*. 9th ed. Boston: Addison-Wesley.
- Sutedi, A., 2011. *Hukum Perizinan Dalam Sektor Pelayanan Publik*. Jakarta: Sinar Grafika Offset.
- Tutorialspoint, 2016. *MySQL Database Management System*. Madhapur: Pvt.Ltd.
- Yasin, V., 2012. *Rekayasa Perangkat Lunak Berorientasi Objek*. Jakarta: Mitra Wacana Media.